Saliency-driven Experience Replay for Continual Learning

Giovanni Bellitto
University of Catania
giovanni.bellitto@unict.it

Federica Proietto Salanitri
University of Catania
federica.proiettosalanitri@unict.it

Matteo Pennisi
University of Catania
matteo.pennisi@phd.unict.it

Matteo Boschini
University of Modena and Reggio Emilia
matteo.boschini@unimore.it

Lorenzo Bonicelli
University of Modena and Reggio Emilia
lorenzo.bonicelli@unimore.it

Angelo Porrello
University of Modena and Reggio Emilia
angelo.porrello@unimore.it

Simone Calderara
University of Modena and Reggio Emilia
simone.calderara@unimore.it

Simone Palazzo
University of Catania
simone.palazzo@unict.it

Concetto Spampinato
University of Catania
concetto.spampinato@unict.it

Abstract

We present *Saliency-driven Experience Replay* - SER - a biologically-plausible approach based on replicating human visual saliency to enhance classification models in continual learning settings. Inspired by neurophysiological evidence that the primary visual cortex does not contribute to object manifold untangling for categorization and that primordial saliency biases are still embedded in the modern brain, we propose to employ auxiliary saliency prediction features as a modulation signal to drive and stabilize the learning of a sequence of non-i.i.d. classification tasks. Experimental results confirm that SER effectively enhances the performance (in some cases up to about twenty percent points) of state-of-the-art continual learning methods, both in class-incremental and task-incremental settings. Moreover, we show that saliency-based modulation successfully encourages the learning of features that are more robust to the presence of spurious features and to adversarial attacks than baseline methods. Code is available at: https://github.com/perceivelab/SER.

1 Introduction

Humans possess the remarkable capability to keep learning, with limited forgetting of past experience, and to quickly re-adapt to new tasks and problems without disrupting consolidated knowledge. Machine learning, on the contrary, has shown significant limitations when dealing with non-stationary data streams with a limited possibility to replay past examples. The main reason for this shortcoming

38th Conference on Neural Information Processing Systems (NeurIPS 2024).

can be found in the inherent structure, organization and optimization approaches of artificial neural networks, which differ significantly from how humans learn and how their neural connectivity is built when accumulating knowledge over a lifetime. According to the *Complementary Learning Systems (CLS) theory* [46, 33], the human ability to learn effectively may be due to the interplay between two learning processes that originate, respectively, on the hippocampus and on the neocortex. This theory has inspired several continual learning methods [29, 40, 28]. In particular, the recent DualNet method [51] translates CLS concepts into a computational framework for continual learning. Specifically, it employs two learning networks: a *slow learner*, emulating the memory consolidation process happening in the hippocampus through contrastive learning techniques, and a *fast learner*, that aims at adapting current representations to new observations. However, this strategy still appears insufficient for addressing the problem of continual learning, because it starts from the (possibly wrong) assumption that human neural networks directly process visual input with the objective of performing categorization from early vision layers. On the contrary, neurophysiological studies [19, 32] are in near universal agreement that the object manifolds conveyed to primary visual cortex V1 (one of the earliest areas involved in vision) are as tangled as the pixel space. In other words, the neurons of the earliest vision areas do not contribute to object manifold untangling for categorization, but rather enforce luminance and contrast robustness [32]. This suggests that training early neurons with a visual categorization objective — as done not only in DualNet, but in all existing continual learning methods — is in stark contrast to the biological counterparts observed in primates. Moreover, recent studies on the causes of forgetting in artificial neural networks showed that deeper layers (i.e., closer to the output) are less stable in presence of task shifts [53], which is consistent with the hypothesis that earlier layers do not bear specific categorization responsibilities.

Given these premises, it is peculiar that existing bio-inspired continual learning methods tend to ignore all upstream neural processes underlying visual categorization, such as visual saliency processes. Indeed, the ability to select relevant visual information appears to be the hallmark of human/primate cognition. Moreover, recent findings in cognitive neuroscience have shown that the visual attention priorities of human hunter-gatherer ancestors are still embedded in the modern brain [48]: humans pay attention faster to animals than to vehicles, although we now see more vehicles than animals. This primordial saliency bias embedded in human brains suggests that the neuronal circuits of the ventral visual pathway are somehow inherited, as a form of genetic legacy from ancestral experience, and tend to remain stable over time — thus not subject to forgetting, though we have long stopped hunting to survive. Interestingly, we observed the same **forgetting-free** behavior for saliency prediction on artificial neural networks. Fig. 1 shows the trend of the *similarity* [10] metric for a saliency prediction model trained in a continual learning scenario, and compares it to the accuracy of a classification model under the same settings. While classification accuracy drops as the classifier learns new classes, the saliency metric remains stable, and even slightly improves.

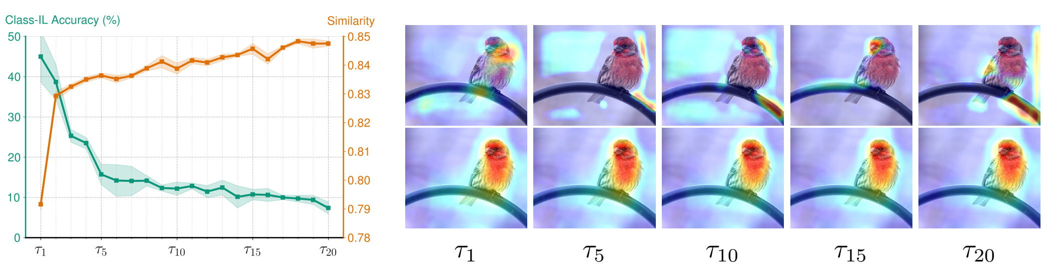

Figure 1: **Comparison of Forgetting-Free Saliency Prediction vs. Catastrophic Forgetting in Classifiers and Activation Maps in Continual Learning Scenarios**. *(Left figure)*: The saliency accuracy (measured by the *similarity* [10] score) of a saliency predictor trained in a continual learning setting improves as more tasks are introduced, while the classification accuracy of a continual classifier degrades over time, indicating that saliency detection remains i.i.d. even with non-i.i.d. data. *(Right figure)*: The top row shows activation maximization maps via GradCAM, which are prone to catastrophic forgetting due to their dependence on the classifier. In contrast, the bottom row shows saliency maps produced by the predictor, which remain stable and consistent over time.

From this observation, in this paper we propose *SER*, a *Saliency-driven Experience Replay* strategy that employs visual saliency prediction [6] to drive the learning of a sequence of classification tasks in a continual learning setting. To emulate what has been observed in primates, where visual saliency modulates the firing rate of neurons that represent the attended stimulus at different stages of

visual processing [63, 45], SER adopts a two-branch model: one branch performs visual saliency prediction [37, 27, 20], and its responses modulate the features learned by a paired classification model in the second branch.
While the SER strategy stands out in its approach, it's important to note a similar category of methodologies that utilize attribution maps (e.g., computed via GradCAM), also known as attention maps, as a distilled form of classifier knowledge for future replay [61, 18, 22, 59, 3]. However, **saliency prediction maps are significantly different from attribution maps**. Indeed, attribution maps elucidate the inner workings of DNNs by highlighting relevant input features for predictions and as such they suffer catastrophic forgetting (as shown in Fig. 1), while saliency maps, rooted in neuroscience and human visual processing, aim to emulate how humans perceive and prioritize visual information, and, most importantly, they are forgetting-free.

SER is model-agnostic and can be used in combination to any continual learning method. We demonstrate that saliency modulation positively impacts classification performance in online continual learning settings, leading to a significant gain in accuracy (up to 20 percent points) w.r.t. baseline methods. We further demonstrate the usefulness of saliency modulation on different benchmarks (including a challenging one that tackles fine-grained classification) and substantiate our claims through a set of ablation studies. We finally show that saliency modulation, besides being biologically plausible, leads to learn saliency-modulated features that are more robust to the presence of spurious features and to adversarial attacks.

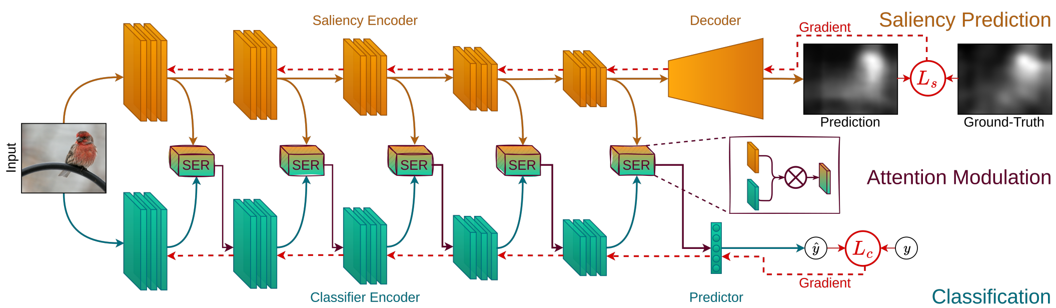

Figure 2: **Architecture of the proposed Saliency-driven Experience Replay (SER) strategy.** The classification backbone is paired with a saliency prediction network that, given its capability of being forgetting-free, aims at adjusting the learned classification features in order to mitigate overall forgetting.

2 Related Work

Continual Learning (CL) [47, 16, 49] addresses the problem of *catastrophic forgetting* in neural networks, wherein they tend to lose previously acquired knowledge when faced with shifts in input data distribution. Various solutions have been proposed to address this, including the incorporation of regularization terms [31, 74], specific architectural designs [60, 44], and rehearsal of previously encountered data points [57, 55, 9]. However, the application of these solutions to real-world scenarios is challenging due to evaluations often being based on unrealistic benchmarks [1, 65]. *Online Continual Learning* (OCL) [43] addresses this challenge by limiting multiple epochs on the input stream, reflecting the realistic assumption that data points encountered in real-world settings occur only once. To address this challenge, many strategies adopt a replay approach [54, 57]. Some focus on memory management: GSS [2] optimizes the basic rehearsal formula to store maximally informative samples, while HAL [14] identifies synthetic replay data points maximally affected by forgetting. CoPE [15] employs class prototypes for gradual evolution of the shared latent space, while ER-ACE [11]adjusts the cross-entropy loss asymmetrically to minimize task imbalance. Our proposal adopts a remarkably different approach w.r.t. these classes of methods, in that we take inspiration from cognitive neuroscience theory of learning and exploiting the features of a conjugate forgetting-free task (i.e., saliency prediction) to modulate the responses of our OCL model. Doing so produces a stabilizing effect on our model and makes it more resilient to forgetting.

An approach similar in the spirit to ours is [39] that leverages saliency prediction for exemplar-free class incremental learning. To compensate for the absence of past task data, this methods relies on a pre-trained saliency detector, which remains frozen throughout the learning process, providing guidance for attribution maps of the classification backbone. Consequently, it tackles the challenge of forgetting by employing a pre-trained backbone to constrain feature drift. In contrast, SER operates on a dynamic framework where the visual saliency network is continuously trained, showcasing remarkable resistance to forgetting, while concurrently modulating the drift of classification features. This approach offers a more flexible visual saliency-classification paradigm that adapts to any dataset without external dependencies, as opposed to [39], which requires the use of a pre-trained saliency detector trained on the same data distribution as the target data.

Another approach, similarly inspired by cognitive theories, is DualNet [51], which employs two networks that loosely emulate how slow and fast learning work in humans. However, DualNet employs contrastive learning on the slow network (the earliest layers of the model), while it seems that object-identifying transformations happens later in the human visual system [19, 32]. Our results, reported later, substantiate the suitability of our choice to use low-level processes, such as saliency prediction, to drive continual learning tasks, rather than contrastive learning or classification pre-training techniques as, respectively, in DualNet and TwF [8].

Though the concept of utilizing saliency prediction maps in online continual learning is relatively new, recent trends have shown promising advancements in mitigating forgetting by encouraging models to recall evidence for past decisions, stored as activation maps [22]. Specifically, [22, 59, 3] employ attribution methods, such as Gradient-weighted Class Activation Mapping (Grad-CAM) [61], to compute and store visual model explanations for each sample (or parts thereof) in the buffer and ensures model consistency with previous decisions during the training phase. Similarly, Dhar [18] adopts Grad-CAM, but it does not store any information, it employs knowledge distillation on the activation maps across consecutive tasks. However, as presented in the introduction, there is a fundamental distinction between saliency maps and activation maps with the latter being subject to forgetting, while the former not (Fig. 1).

Finally, our approach diverges from the recent trend in the continual learning (CL) field, which primarily employs foundation models (mostly Vision Transformers, ViTs) and focuses on learning prompts to mitigate forgetting [68, 67, 24, 62]. The main limitation of these methods is that they are restricted to transformer-based architectures. In contrast, our strategy does not rely on any specific model type, thereby enhancing its potential impact on real-world applications.

3 Method

3.1 Online Continual Learning

Following the recent literature, we pose OCL as a supervised image classification problem with an online non-i.i.d. stream of data, where each training sample is only seen once. Although our saliency-driven modulation does not require the presence or knowledge of *task boundaries*, in this formulation and in our experiments we assume that these are given, to the benefit of any baseline method enhanced by the proposed extension. More formally, let $\mathcal{D} = \{\mathcal{D}_1, \dots, \mathcal{D}_T\}$ be a sequence of data streams, where each pair $(\mathbf{x}, y) \sim \mathcal{D}_i$ denotes a data point $\mathbf{x} \in \mathcal{X}$ with the corresponding class label $y \in \mathcal{Y}$; the sample distributions (in terms of both the data point and the class label) differ between separate streams $\mathcal{D}_i$ and $\mathcal{D}_j$ — the sets of class labels in each stream are disjoint, though both belong to the same domain $\mathcal{Y}$. Given a classifier $f : \mathcal{X} \to \mathcal{Y}$, parameterized by $\boldsymbol{\theta}$, the objective of OCL is to train f on $\mathcal{D}$, organized as a sequence of T tasks $\{\tau_1, \dots, \tau_T\}$, under the constraint that, at a generic task τ_i, the model receives inputs sampled from the corresponding data distribution, i.e., $(\mathbf{x}, y) \sim D_i$, and sees each sample only once during the whole training procedure. The classification model may optionally keep a limited *memory buffer* $\mathbf{M}$ of past samples, to reduce forgetting of features from previous tasks. The model update step between tasks can be summarized as:

$$\langle f, \boldsymbol{\theta}_{i-1}, \mathcal{D}_{i-1}, \mathbf{M}_{i-1} \rangle \to \langle f, \boldsymbol{\theta}_i, \mathbf{M}_i \rangle \tag{1}$$

where $\boldsymbol{\theta}_i$ and $\mathbf{M}_i$ represent the set of model parameters and the buffer at the end of task τ_i, respectively. For methods that do not exploit buffer, $\mathbf{M}_i = \varnothing, \forall i$.

The training objective is to optimize a classification loss over the sequence of tasks (without losing accuracy on past tasks) by the model instance at the end of training:

$$\arg\min_{\boldsymbol{\theta}_T} \sum_{i=1}^{T} \mathbb{E}_{(\mathbf{x},y)\sim\mathcal{D}_i}\left[\mathcal{L}\Big(f\left(\mathbf{x};\boldsymbol{\theta}_T\right), y\Big)\right] \tag{2}$$

where $\mathcal{L}$ is a generic classification loss (e.g., cross-entropy), which a continual learning model attempts to optimize while accounting for model *plasticity* (the capability to learn current task data) and *stability* (the capability to retain knowledge of previous tasks) [47].

3.2 SER: Saliency-driven Experience Replay

Our method is grounded on the neurophysiological evidence that attention-driven neuronal firing rate modulation is multiplicative and the scaling of neuronal responses depends on the similarity between a neuron's preferred stimulus and the attended feature [63, 45]. This hypothesis is translated into a general artificial neural architecture, where we emulate the process of human selective attention through a visual saliency prediction network [6] whose activations modulate, through multiplication, neuron activations of a paired classification network at different stages of visual processing. Formally, let $S : \mathcal{X} \to \mathcal{S}$ be a saliency prediction network, where $\mathcal{X}$ is the space of input images and $\mathcal{S}$ the space of output saliency maps. Generally, if $\mathcal{X} = \mathbb{R}^{3\times H\times W}$ for RGB images, then $\mathcal{S} = \mathbb{R}^{H\times W}$, where each location of a map $\mathbf{s} \in \mathcal{S}$ measures the *saliency* of the corresponding pixel in the RGB space. We assume that S can be decomposed into two functions, an encoder $E : \mathcal{X} \to \mathcal{H}$ and a decoder $D : \mathcal{H} \to \mathcal{S}$, such that $S\left(\mathbf{x}\right) = D\left(E\left(\mathbf{x}\right)\right)$, for $\mathbf{x} \in \mathcal{X}$. Then, given an online continual learning problem with data stream $\mathcal{D}$ and set of classes $\mathcal{Y}$, let $C : \mathcal{X} \to \mathcal{Y}$ be a classification network, such that C and the saliency encoder E share the same architecture (with independent parameters). An illustration of the proposed architecture is shown in Fig. 2.

At training time, both S and C observe the same data stream, from which pairs $(\mathbf{x}, y)$ of input data and class label are iteratively sampled. Through the use of an external *saliency oracle*, we extend each data sample to a triple $(\mathbf{x}, y, \mathbf{s})$, where $\mathbf{s}$ is the target saliency map associated to $\mathbf{x}$. The oracle can be either a set of ground-truth maps, when available, or *pseudo-labels* provided as the output of a pre-trained saliency predictor (unrelated to S). We therefore proceed to optimize a multi-objective loss function $\mathcal{L} = \mathcal{L}_s + \lambda\mathcal{L}_c$, with λ being a weighing hyperparameter. Loss term $\mathcal{L}_s$ is computed on the output of saliency predictor S, and compares the estimated saliency map $S(\mathbf{x})$ with the target $\mathbf{s}$ by means of the Kullback-Leibler divergence (commonly employed as a saliency prediction objective [10, 20, 4, 69, 26]):

$$\mathcal{L}_s = \sum_i s_i \log\left(\frac{s_i}{S_i(\mathbf{x}) + \epsilon} + \epsilon\right) \tag{3}$$

with s_i and $S_i(\mathbf{x})$ iterating over map pixels in $\mathbf{s}$ and $S(\mathbf{x})$, respectively. Loss term $\mathcal{L}_c$ encodes a generic online continual learning objective, as introduced in Eq. 2. As the proposed approach is method-agnostic, details on the formulation of $\mathcal{L}_c$ may vary.

In order to enforce selective attention-driven modulation of classification neuronal activations, we leverage the architectural identity of saliency prediction encoder E and classifier C to alter the feedforward pass of the latter, by multiplying pre-activation features in C by the corresponding features in E, before applying a non-linearity and feeding them to the next layer of the network. Formally, let us assume that the C and E networks consist of a sequence of layers $\{l_1, l_2, \dots, l_L\}$. Without loss of generality, let each layer l_i compute its output as $\mathbf{z}_i = \sigma\left(\mathbf{W}_i\mathbf{z}_{i-1}\right)$, with σ being an activation function, $\mathbf{W}_i$ the network-specific layer parameters (i.e., not shared between E and C) and $\mathbf{z}_{i-1}$ the output of the previous layer (or the network's input $\mathbf{x}$, if appropriate). Then, let us distinguish between features $\mathbf{z}_i^{(s)}$ and $\mathbf{z}_i^{(c)}$, respectively representing the output of layer l_i by the saliency prediction encoder S and the classifier C. We apply saliency-driven modulation by modifying the computation of $\mathbf{z}_i^{(c)}$ as follows:

$$\mathbf{z}_i^{(c)} = \sigma\Big(\mathbf{W}_i^{(c)}\left(\mathbf{z}_{i-1}^{(c)} \odot \mathbf{z}_{i-1}^{(s)}\right)\Big) \tag{4}$$

where $\odot$ denotes the Hadamard product. Intuitively, the proposed approach encourages the classification model to attend to "salient" features of the input, where the concept of *saliency* is generalized from the pixel space to hidden representations. It is important to note that, at training time, gradient descent optimization of $\mathcal{L}_c$ would also affect on the saliency encoder E. This is undesirable, as we previously showed (see Fig. 1) that saliency features are robust to task shifts, unlike classification features: hence, in order to guarantee this property, we stop the gradient flow from $\mathcal{L}_c$ to parameters in E, and use it to update the parameters of classifier C only.

In the above formulation, we assumed the presence of a classification network with fully-connected layers; however, our method can be applied in an agnostic manner to any method employing, at least in part, a feature extractor implemented as a neural network. As such, the proposed method can be equally applied, for instance, both to end-to-end classification models (e.g., DER++ [9]) and to approaches with a neural backbone that computes class-representative prototypes (e.g., CoPE [15]).

4 Performance Analysis

4.1 Experimental setup

Benchmarks. We build two OCL benchmarks by taking image classification datasets and splitting their classes equally into a series of disjoint tasks:

- **Split Mini-ImageNet** [66, 13, 21, 17] that includes 100 classes from ImageNet, allowing for a longer task sequence. For each class, 500 images are used for training and 100 for evaluation.
- **Split FG-ImageNet**[1] [58] is a benchmark for fine-grained image classification that we use to test CL methods on a more challenging task than traditional ones. It includes 100 classes of animals extracted from ImageNet, belonging to 7 different species, reducing inter-class variability and leading to harder tasks. Each class contains 500 samples for training and 50 for evaluation.

For both datasets, images are resized to 288×384 pixels and split into twenty 5-way tasks.

Baseline methods. We evaluate the contribution of the SER strategy when paired to a classification network trained using several state-of-the-art continual learning approaches, including rehearsal and non-rehearsal methods:

- **DER++** [7]: a seminal work that combines rehearsal and knowledge distillation strategies for supporting model plasticity while limiting forgetting.
- **ER-ACE** [11]: a variant of Experience Replay [54, 57] which aims to prevent imbalances due to the simultaneous optimization of the current and past tasks by selectively masking softmax outputs.
- **CoPE** [15]: a prototype-based classifier with experience replay, whose careful update scheme prevents sudden disruptions in the latent space during incremental learning.
- **LwF** [36]: a non-rehearsal method that enforces a model to preserve outputs of past model instances on new samples to limit forgetting.
- **oEWC** [30]: a non-rehearsal method that mitigates forgetting by selectively limiting the changes on weights that are most informative of past tasks.

Implementation details. We apply the SER strategy at five feature modulation points of ResNet-18's architecture, namely, the outputs of the first convolutional block and of the four main residual blocks. In compliance with online learning, all models are trained for a single epoch, using SGD as optimizer, with a fixed batch size of 8 both for the input stream and the replay buffer. Rehearsal methods are evaluated with three different sizes of the memory buffer (1000, 2000 and 5000). When applying SER, besides each method's specific training objective, we also optimize the saliency prediction loss $\mathcal{L}_s$ from Eq. 3, with $\lambda = 1$. Saliency is estimated using DeepGaze IIE network [37] as oracle.

When using SER, classifier C and saliency predictor S are identical ResNet-18 architectures, followed — respectively — by a linear classification layer and a saliency map decoder (additional details are provided in the supplementary materials). While C is trained from scratch, we employ a pre-trained saliency predictor S, consistently with neuroscience evidence showing that humans have selective attention already embedded in the brain [48]. For a fair comparison, in all our experiments feature extraction backbones of baseline methods are initialized to the same pre-trained weights as S (except where explicitly stated). Care was taken to ensure that the set of OCL classes $\mathcal{C}$ did not semantically overlap with pre-training data, to prevent any contamination from the saliency predictor to the classification task. Specifically, S was pre-trained for 20 epochs on a subset of 100 ImageNet classes (disjoint from our two main benchmark datasets), using DeepGaze IIE as oracle. No class label information was used at this stage. All experiments were conducted on a workstation with an 24-core

[1] https://www.kaggle.com/datasets/ambityga/imagenet100

CPU, 500GB RAM, and an NVIDIA A100 GPU (40GB VRAM). Results are computed using the Mammoth framework [9].

Metrics and evaluation. As a primary metric of OCL model performance, we report the *final average accuracy* as $\frac{1}{T}\sum_{i=1}^{T} a_i^T$, where a_i^T is the accuracy of the final model on the test set of task τ_i. Accuracy a_i^T can be computed in a *Class-Incremental Learning* (*Class-IL*) or in a *Task-Incremental Learning* (*Task-IL*) setting. In the latter, we assume that a task identifier is provided to the model at inference time, simplifying the problem by restricting the set of class predictions for a given sample. While Task-IL is often depicted as a trivial scenario in recent literature [23, 64, 2], we emphasize its usefulness, as it isolates the effect of within-task forgetting from the model's bias towards the currently learned classes [71, 25, 7]. In the paper, we mainly report results in Class-IL, while the results in Task-IL setting are given in the supplementary materials. Results are reported in term of mean and standard deviation over five different runs.

4.2 Results

We first evaluate the contribution that saliency-driven modulation provides to state-of-the-art OCL baselines. For each method, we compute Class-Incremental accuracy and compare to those obtained when integrating SER, as described in Sec. 3. Since our strategy foresees two paired networks for classification and saliency prediction, we also compare with similar multi-branch CL baselines:

- **DualNet** [51], mentioned in Sec. 1, employs a dual-backbone architecture to decouple incremental classification (by a *fast learner*) from self-supervised representation learning [73] (by a *slow learner*). We adapt SER to DualNet by replacing the slow learner and its training objective with our saliency prediction backbone, forcing the fast learner to use saliency features for classification.
- **TwF** [8] employs a frozen pre-trained classification backbone to stabilize the learning of Class-Incremental features, by means of an attention mechanism. To enable SER, the pre-trained classification backbone and the feature distillation strategy are replaced with the saliency encoder, and the features of the two backbones are combined through multiplication, as described in Sec. 3.

Results are reported in Table 1, showing a pattern of enhanced performance when integrating SER up to 20 percent points. In terms of comparison against two-paired networks, integrating SER outperforms both of them, suggesting that controlling learning through saliency leads to better representation for classification than, for instance, contrastive learning (as done in DualNet) or feature attention with a pre-trained backbone (as in TwF)[2]. This is inline with cognitive neuroscience [19, 35], for which object identity-preservation, that also involves contrastive learning, happens mostly at later layers (e.g., IT neurons), while selective attention (through visual saliency) acts during the whole categorization process. Results for non-rehearsal methods are reported in the supplementary materials.

4.3 Ablation Study

The proposed strategy is grounded on cognitive neuroscience literature, according to which selective attention modulates neuronal responses of all layers involved in the categorization process, in a multiplicative fashion. Our next experiments are meant to assess whether this hypothesis (i.e., feature modulation through multiplication for all classification layers) is optimal also for artificial neural networks, or if other integration modalities of saliency information may be equally effective. We thus compare our SER strategy with the following baselines, all exploiting saliency information in different ways:

- **Saliency-based input modulation (SIM)**: the input image is multiplied by the corresponding estimated saliency map (thus highlighting salient regions only).
- **Saliency as additional input (SAI)**: we modify the classification network to receive as input a 4D data tensor, with the saliency map concatenated to RGB channels.
- **Learning saliency-based modulation (LSM)**: rather than multiplying classification features $\mathbf{z}_{i-1}^{(c)}$ and saliency features $\mathbf{z}_{i-1}^{(s)}$ (see Eq. 4), we feed them to convolutional layer with 1×1 kernel to produce $\mathbf{z}_i^{(c)}$, and let the model learn the corresponding parameters.

[2]We could not run TwF with buffer size of 5000, due to excessive computing requirements.

Table 1: **Class-Incremental accuracy of SOTA rehearsal-based methods** with and without SER.

Model	**Split Mini-ImageNet**			**Split FG-ImageNet**		
Joint	14.79±1.17			9.06±1.07		
Fine-tune	3.43±0.35			2.43±0.81		
Buffer size	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**
DER++	14.95±3.11	12.82±4.97	14.58±2.55	8.08±1.54	8.27±1.72	9.20±0.86
↪**SER**	**19.13**±1.62	**22.92**±2.25	**25.35**±2.56	**11.71**±2.36	**12.97**±1.62	**13.73**±1.95
ER-ACE	20.86±3.69	24.93±3.20	26.31±5.22	14.28±0.96	16.45±1.24	18.21±3.45
↪**SER**	**27.48**±2.83	**33.09**±1.28	**35.58**±1.79	**20.03**±3.13	**23.80**±2.11	**28.68**±0.50
CoPE	21.58±1.60	23.58±4.39	24.77±3.56	16.45±1.38	16.81±0.83	17.77±2.02
↪**SER**	**26.66**±2.22	**33.35**±4.67	**45.04**±2.44	**18.17**±2.79	**27.14**±1.62	**34.34**±3.51
	Dual-branch methods					
TwF	23.78±1.67	29.05±2.02	–	15.32±2.59	18.72±1.75	–
↪**SER**	**28.36**±3.72	**35.55**±0.61	–	**20.04**±1.63	**22.54**±2.20	–
DualNet	20.57±0.91	27.41±1.79	32.08±1.55	15.62±1.54	21.04±1.08	22.07±2.08
↪**SER**	**28.58**±1.40	**33.76**±1.21	**36.44**±0.77	**19.48**±0.59	**22.53**±1.56	**24.83**±2.01

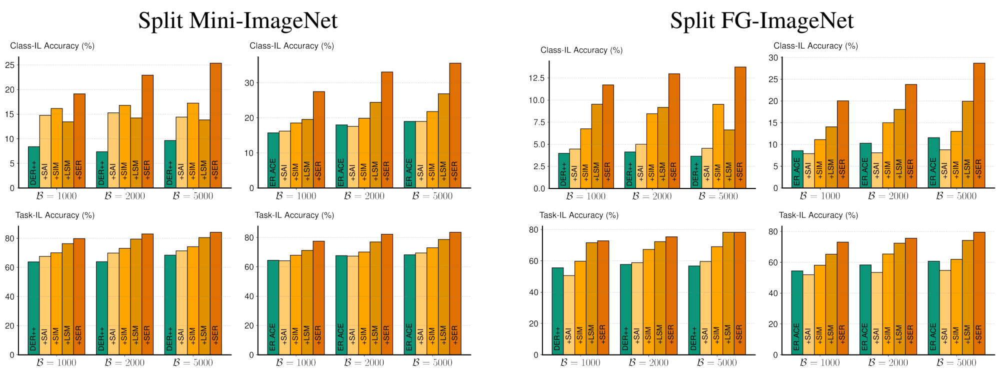

Figure 3: **Comparison of SER to alternative saliency integration strategies**. **SIM** modulates input images by saliency maps. **SAI** provides saliency maps as an additional input channel to the classification network. **LSM** merges classification and saliency features through a learnable convolutional layer.

Fig. 3 reports the results of this analysis, using DER++ and ER-ACE as baseline methods, and clearly indicates the superiority the SER strategy to other saliency integration variants. However, it is interesting to note that saliency helps classification performance in all cases, demonstrating its usefulness for continual learning tasks. We argue that this is due to the intrinsic nature of saliency prediction, which we found to be i.i.d. with respect to the data stream.

We then investigate whether the impact of selective-driven modulation is uniform across the backbone layers. To this aim, we define a positional binary coding scheme, controlling the application of the SER strategy at the predefined points of the network (see Sect. 4.1): if position i of the coding scheme is 1, then the i-th feature modulation point is enabled, i.e., features from the i-th block of the classification network are multiplied by the features of the i-th block of the saliency network. Results are reported in Table 2 for both DER++ and ER-ACE, and indicate that the best strategy is to modulate the features of all classification layers through the corresponding saliency ones, similarly to what neurophysiological evidence reports [63, 45].

4.4 Model Robustness

We finally assess the robustness of the SER strategy to *spurious features* and *adversarial attacks*. Spurious features are information that correlates well with labels in training data but not in test data

Table 2: **Performance comparison when applying SER to DER++ and ER-ACE** at different layers of the ResNet-18 backbone, with buffer size 2000 (Class-IL).

	Split Mini-ImageNet		Split FG-ImageNet	
SER Scheme	**DER++**	**ER-ACE**	**DER++**	**ER-ACE**
1 1 1 0 0	12.97±2.62	23.72±0.77	6.54±0.67	18.08±0.96
1 1 1 1 0	17.46±1.02	26.44±2.33	8.77±1.45	16.55±2.55
1 1 1 1 1	**22.92**±2.25	**33.09**±1.28	**12.97**±1.62	**23.80**±2.11

(e.g., in a classification task between birds and dogs, training with yellow birds and black dogs only), leading to low generalization [34]. This effect is exacerbated in continual learning settings, where the covariate shift between train data and test data increases as new tasks come in. Thus, we measure to what extent our SER strategy can mitigate the tendency of learning methods to exploit spurious features to solve classification tasks. We crafted an ad-hoc benchmark consisting of ten classes from ImageNet. For each class, we added a class signature for training images, leaving the test images unaltered. In detail, we modified each training image by increasing the brightness of all pixels by a class-dependent offset, computed as $5(c+1)$ (in a 0-255 brightness range), where c is a numeric class label. We then define five continual learning tasks with two classes each. We then compare ER-ACE to the corresponding SER-enabled variant and ground its performance with the one obtained when it is trained with original images (i.e., without enforcing spurious features in the data). Results in Table 3 show that SER effectively limits the possibility for the classifier to use spurious features, resulting in a more robust and generalizing model. The drop of performance (about 22 percent points) observed between training with the original data and training with data biased by spurious features is almost completely recovered when SER is used.

Finally, we evaluate the robustness of SER against adversarial perturbations of the input space. To this aim, we apply the Projected Gradient Descent (PGD) attack [42] with different ε values (determining the strength of the attack) and compare the average performance drop experienced by ER-ACE, in its original version and when combined with SER. We conduct the evaluation on both Split Mini-ImageNet and Split FG-ImageNet, repeating each experiment three times. As shown in Figure 4, SER considerably improves model stability, counteracting perturbations by regularizing classification features with saliency ones.

Supplementary materials include additional experiments: performance in Task-IL settings, results for buffer-free methods, effect of pre-training on a pre-text task for the classifier and saliency predictor backbones, cost analysis showing training and inference times of our approach compared to existing methods.

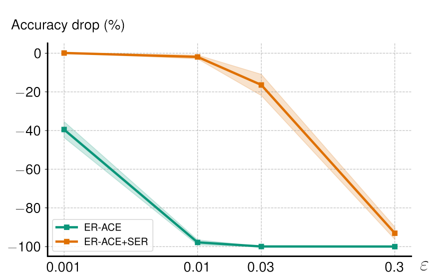

Figure 4: **Robustness to adversarial attacks**. ER-ACE baseline drops even with small attacks, while SER significantly enhances robustness.

Method	Class-IL	Task-IL
ER-ACE	50.07±3.88	86.77±1.63
ER-ACE$^{\mathcal{SF}}$	28.46±3.46	74.40±4.37
↪**SER**	**44.08**±3.67	**83.04**±3.06

Table 3: **Effect of the SER strategy in the presence of spurious features**. The $\mathcal{SF}$ apex shows training on a biased dataset with spurious features.

5 Conclusion

We presented SER, a biologically-inspired saliency-driven modulation strategy for online continual learning, which regularizes classification features using visual saliency, effectively reducing forgetting. The proposed approach, grounded on neurophysiological evidence, significantly improves performance of state-of-the-art OCL methods, and has been shown to be superior to other multi-branch solutions, either biologically-inspired (e.g., DualNet) or based on attention mechanisms (e.g., TwF). Our results confirm that adapting neurophysiological processes into current machine learning techniques is a promising direction to bridge the gap between humans and machines.

Limitations and future works. In this work, we introduce the use of saliency maps as auxiliary knowledge to mitigate forgetting in continual learning. This involves pre-training our saliency predictor with an oracle, which could be in the form of either ground-truth maps or an external model generating pseudo-labels. High-quality input images are necessary for producing meaningful saliency maps, thus, datasets like CIFAR10/100 cannot be employed due to their lower resolution.
Although SER is model-agnostic, its formulation necessitates that the saliency encoder and the classifier share identical architectures. To apply this to heterogeneous networks, we will explore defining or learning mappings between activations at different network stages.

Finally, our finding that saliency prediction is *i.i.d.* with respect to classification distribution shifts opens the door to investigating whether other low-level visual tasks share this property.

6 Acknowledgements

G. Bellitto, F. Proietto Salanitri and C. Spampinato acknowledge financial support from PNRR MUR project PE0000013-FAIR. M. Pennisi is a PhD student enrolled in the National PhD in Artificial Intelligence, cycle XXXVII, course on Health and life sciences, organized by Università Campus Bio-Medico di Roma.

References

[1] Rahaf Aljundi, Klaas Kelchtermans, and Tinne Tuytelaars. Task-free continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11254–11263, 2019.

[2] Rahaf Aljundi, Min Lin, Baptiste Goujaud, and Yoshua Bengio. Gradient Based Sample Selection for Online Continual Learning. In *Advances in Neural Information Processing Systems*, 2019.

[3] Guangji Bai, Qilong Zhao, Xiaoyang Jiang, and Liang Zhao. Saliency-guided hidden associative replay for continual learning. In *Associative Memory & Hopfield Networks in 2023*, 2023. URL https://openreview.net/forum?id=Fhx7nVoCQW.

[4] Giovanni Bellitto, Federica Proietto Salanitri, Simone Palazzo, Francesco Rundo, Daniela Giordano, and Concetto Spampinato. Hierarchical domain-adapted feature learning for video saliency prediction. *International Journal of Computer Vision*, 129:3216–3232, 2021.

[5] Ari S Benjamin, David Rolnick, and Konrad Kording. Measuring and regularizing networks in function space. In *International Conference on Learning Representations Workshop*, 2019.

[6] Ali Borji. Saliency prediction in the deep learning era: Successes, limitations, and future challenges, 2018. URL https://arxiv.org/abs/1810.03716.

[7] Matteo Boschini, Lorenzo Bonicelli, Pietro Buzzega, Angelo Porrello, and Simone Calderara. Class-incremental continual learning into the extended der-verse. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2022.

[8] Matteo Boschini, Lorenzo Bonicelli, Angelo Porrello, Giovanni Bellitto, Matteo Pennisi, Simone Palazzo, Concetto Spampinato, and Simone Calderara. Transfer without forgetting. In *European Conference on Computer Vision*, 2022.

[9] Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark Experience for General Continual Learning: a Strong, Simple Baseline. In *Advances in Neural Information Processing Systems*, 2020.

[10] Zoya Bylinskii, Tilke Judd, Aude Oliva, Antonio Torralba, and Frédo Durand. What do different evaluation metrics tell us about saliency models? *IEEE transactions on pattern analysis and machine intelligence*, 2018.

[11] Lucas Caccia, Rahaf Aljundi, Nader Asadi, Tinne Tuytelaars, Joelle Pineau, and Eugene Belilovsky. New Insights on Reducing Abrupt Representation Change in Online Continual Learning. In *International Conference on Learning Representations Workshop*, 2022.

[12] Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient Lifelong Learning with A-GEM. In *International Conference on Learning Representations Workshop*, 2019.

[13] Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K Dokania, Philip HS Torr, and Marc'Aurelio Ranzato. On tiny episodic memories in continual learning. In *International Conference on Machine Learning Workshop*, 2019.

[14] 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*, 2021.

[15] Matthias De Lange and Tinne Tuytelaars. Continual prototype evolution: Learning online from non-stationary data streams. In *IEEE International Conference on Computer Vision*, 2021.

[16] Matthias De Lange, 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.

[17] Mohammad Mahdi Derakhshani, Xiantong Zhen, Ling Shao, and Cees Snoek. Kernel continual learning. In *International Conference on Machine Learning*, 2021.

[18] Prithviraj Dhar, Rajat Vikram Singh, Kuan-Chuan Peng, Ziyan Wu, and Rama Chellappa. Learning without memorizing. In *IEEE Conference on Computer Vision and Pattern Recognition, CVPR*, pages 5138–5146, 2019.

[19] J. J. DiCarlo, D. Zoccolan, and N. C. Rust. How does the brain solve visual object recognition? *Neuron*, 2012.

[20] Richard Droste, Jianbo Jiao, and J Alison Noble. Unified image and video saliency modeling. In *European Conference on Computer Vision*, 2020.

[21] Sayna Ebrahimi, Franziska Meier, Roberto Calandra, Trevor Darrell, and Marcus Rohrbach. Adversarial continual learning. In *Proceedings of the European Conference on Computer Vision*, 2020.

[22] 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. *Applied AI Letters*, 2021.

[23] Sebastian Farquhar and Yarin Gal. Towards Robust Evaluations of Continual Learning. In *International Conference on Machine Learning Workshop*, 2018.

[24] Qiankun Gao, Chen Zhao, Yifan Sun, Teng Xi, Gang Zhang, Bernard Ghanem, and Jian Zhang. A unified continual learning framework with general parameter-efficient tuning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11483–11493, October 2023.

[25] Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2019.

[26] Feiyan Hu, Simone Palazzo, Federica Proietto Salanitri, Giovanni Bellitto, Morteza Moradi, Concetto Spampinato, and Kevin McGuinness. Tinyhd: Efficient video saliency prediction with heterogeneous decoders using hierarchical maps distillation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, 2023.

[27] Ming Jiang, Shengsheng Huang, Juanyong Duan, and Qi Zhao. Salicon: Saliency in context. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1072–1080, 2015.

[28] Ronald Kemker and Christopher Kanan. Fearnet: Brain-inspired model for incremental learning. *arXiv preprint arXiv:1711.10563*, 2017.

[29] J. Kirkpatrick, R. Pascanu, N. Rabinowitz, J. Veness, G. Desjardins, A. A. Rusu, K. Milan, J. Quan, T. Ramalho, A. Grabska-Barwinska, D. Hassabis, C. Clopath, D. Kumaran, and R. Hadsell. Overcoming catastrophic forgetting in neural networks. *Proc Natl Acad Sci U S A*, 114(13):3521–3526, Mar 2017.

[30] James Kirkpatrick, Razvan Pascanu, Neil Rabinowitz, Joel Veness, Guillaume Desjardins, Andrei A. Rusu, Kieran Milan, John Quan, Tiago Ramalho, Agnieszka Grabska-Barwinska, Demis Hassabis, Claudia Clopath, Dharshan Kumaran, and Raia Hadsell. Overcoming catastrophic forgetting in neural networks. *Proc. of the National Academy of Sciences*, 2017.

[31] 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*, 2017.

[32] A. Kohn. Visual adaptation: physiology, mechanisms, and functional benefits. *J Neurophysiol*, 2007.

[33] D. Kumaran, D. Hassabis, and J. L. McClelland. What Learning Systems do Intelligent Agents Need? Complementary Learning Systems Theory Updated. *Trends Cogn Sci*, 20(7):512–534, Jul 2016.

[34] Timothée Lesort. Continual feature selection: Spurious features in continual learning, 2022.

[35] N. Li, D. D. Cox, D. Zoccolan, and J. J. DiCarlo. What response properties do individual neurons need to underlie position and clutter "invariant" object recognition? *J Neurophysiol*, 102(1):360–376, Jul 2009.

[36] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

[37] Akis Linardos, Matthias Kümmerer, Ori Press, and Matthias Bethge. Deepgaze iie: Calibrated prediction in and out-of-domain for state-of-the-art saliency modeling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 12919–12928, 2021.

[38] Xialei Liu, Jiang-Tian Zhai, Andrew D Bagdanov, Ke Li, and Ming-Ming Cheng. Task-adaptive saliency guidance for exemplar-free class incremental learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23954–23963, 2024.

[39] Xialei Liu, Jiang-Tian Zhai, Andrew D. Bagdanov, Ke Li, and Mingg-Ming Cheng. Task-adaptive saliency guidance for exemplar-free class incremental learning. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024.

[40] David Lopez-Paz and Marc'Aurelio Ranzato. Gradient episodic memory for continual learning. In *Proceedings of the 31st International Conference on Neural Information Processing Systems*, NIPS'17, page 6470–6479, Red Hook, NY, USA, 2017. Curran Associates Inc. ISBN 9781510860964.

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

[42] Aleksander Madry, Aleksandar Makelov, Ludwig Schmidt, Dimitris Tsipras, and Adrian Vladu. Towards deep learning models resistant to adversarial attacks. In *International Conference on Learning Representations*, 2018.

[43] Zheda Mai, Ruiwen Li, Jihwan Jeong, David Quispe, Hyunwoo Kim, and Scott Sanner. Online continual learning in image classification: An empirical survey. *Neurocomputing*, 2022.

[44] 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*, 2018.

[45] J. C. Martinez-Trujillo and S. Treue. Feature-based attention increases the selectivity of population responses in primate visual cortex. *Curr Biol*, 14(9):744–751, May 2004.

[46] J. L. McClelland, B. L. McNaughton, and R. C. O'Reilly. Why there are complementary learning systems in the hippocampus and neocortex: insights from the successes and failures of connectionist models of learning and memory. *Psychol Rev*, 102(3):419–457, Jul 1995.

[47] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. *Psychology of learning and motivation*, 1989.

[48] Joshua New, Leda Cosmides, and John Tooby. Category-specific attention for animals reflects ancestral priorities, not expertise. *Proceedings of the National Academy of Sciences*, 104(42):16598–16603, 2007. doi: 10.1073/pnas.0703913104. URL https://www.pnas.org/doi/abs/10.1073/pnas.0703913104.

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

[50] Federico Pernici, Matteo Bruni, Claudio Baecchi, Francesco Turchini, and Alberto Del Bimbo. Class-incremental learning with pre-allocated fixed classifiers. In *International Conference on Pattern Recognition*, 2021.

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

[52] Ameya Prabhu, Philip HS Torr, and Puneet K Dokania. GDumb: A simple approach that questions our progress in continual learning. In *Proceedings of the European Conference on Computer Vision*, 2020.

[53] Vinay Venkatesh Ramasesh, Ethan Dyer, and Maithra Raghu. Anatomy of catastrophic forgetting: Hidden representations and task semantics. In *International Conference on Learning Representations Workshop*, 2021.

[54] Roger Ratcliff. Connectionist models of recognition memory: constraints imposed by learning and forgetting functions. *Psychological Review*, 1990.

[55] 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*, 2017.

[56] 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 *International Conference on Learning Representations Workshop*, 2019.

[57] Anthony Robins. Catastrophic forgetting, rehearsal and pseudorehearsal. *Connection Science*, 1995.

[58] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, et al. Imagenet large scale visual recognition challenge. *International journal of computer vision*, 115:211–252, 2015.

[59] Gobinda Saha and Kaushik Roy. Saliency guided experience packing for replay in continual learning. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 5273–5283, 2023.

[60] 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*, 2018.

[61] 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. *International Journal of Computer Vision*, 2019.

[62] James Seale Smith, Leonid Karlinsky, Vyshnavi Gutta, Paola Cascante-Bonilla, Donghyun Kim, Assaf Arbelle, Rameswar Panda, Rogerio Feris, and Zsolt Kira. Coda-prompt: Continual decomposed attention-based prompting for rehearsal-free continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 11909–11919, 2023.

[63] S. Treue and J. C. nez Trujillo. Feature-based attention influences motion processing gain in macaque visual cortex. *Nature*, 399(6736):575–579, Jun 1999.

[64] Gido M van de Ven and Andreas S Tolias. Three continual learning scenarios. In *Neural Information Processing Systems Workshops*, 2018.

[65] Gido M van de Ven, Tinne Tuytelaars, and Andreas S Tolias. Three types of incremental learning. *Nature Machine Intelligence*, 2022.

[66] Oriol Vinyals, Charles Blundell, Timothy Lillicrap, Daan Wierstra, et al. Matching networks for one shot learning. In *Advances in Neural Information Processing Systems*, 2016.

[67] Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *Computer Vision – ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXVI*, page 631–648. Springer-Verlag, 2022.

[68] 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 *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 139–149, June 2022.

[69] Ziqiang Wang, Zhi Liu, Gongyang Li, Yang Wang, Tianhong Zhang, Lihua Xu, and Jijun Wang. Spatio-temporal self-attention network for video saliency prediction. *IEEE Transactions on Multimedia*, 2021.

[70] Yujie Wei, Jiaxin Ye, Zhizhong Huang, Junping Zhang, and Hongming Shan. Online prototype learning for online continual learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18764–18774, 2023.

[71] Yue Wu, Yinpeng Chen, Lijuan Wang, Yuancheng Ye, Zicheng Liu, Yandong Guo, and Yun Fu. Large scale incremental learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, 2019.

[72] Michal Zajac, Tinne Tuytelaars, and Gido M van de Ven. Prediction error-based classification for class-incremental learning. In *The Twelfth International Conference on Learning Representations*, 2024.

[73] Jure Zbontar, Li Jing, Ishan Misra, Yann LeCun, and Stéphane Deny. Barlow twins: Self-supervised learning via redundancy reduction. In *International Conference on Machine Learning*, 2021.

[74] Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International Conference on Machine Learning*, 2017.

A Supplementary Materials

A.1 Architectural Details of the Saliency Prediction Network

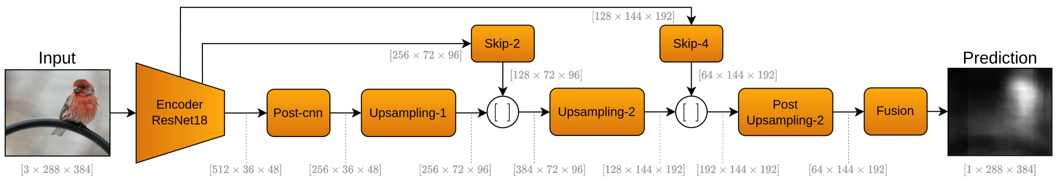

Figure SF-1: Overview of the Saliency Prediction Network used for our experiments

For our experiments we create an *ad-hoc* encoder-decoder saliency prediction network with skip connections. This network uses a ResNet-18 as encoder as to be similar to the paired classifier, thus easing the saliency-based modulation between the two branches.
The saliency decoder is instead broadly inspired by UNISAL[20]. We opted for UNISAL decoder because of the low number of parameters it requires, which leads to a short runtime if compared to other saliency models[3]. In particular, the decoder consists of a stack of pointwise convolutions and deptwhise separable 3×3 convolutions, interleaved with bilinear upsampling blocks until the size of the original input image is recovered, while features from second and third residual blocks of the Encoder are used as skip connections, through two modules named *Skip-2* and *Skip-4*, to fuse features extracted at different abstraction levels. The architecture of the proposed model is illustrated in Fig. SF-1. Essentially, features from the bottleneck are upsampled with a factor $\alpha = 2$ and concatenated with the output of Skip-2 module. The obtained features maps are upsampled again with a factor $\beta = 2$ and concatenated with the output of Skip-4 module, while the number of feature maps is progressively scaled from the original value of 512 to 64. One last 1×1 convolution, followed by an upsampling layer and logistic activation, reduces the feature maps to 1 and the spatial sizes are restored to those of the input image. More details are reported in Table ST-1.

Table ST-1: **Detailed input-output sizes of the Decoder of our Saliency Prediction Network**

Saliency Model: Decoder						
Name	**type**	**kernel/(stride)**	**Batch Norm**	**Activation**	**Input shape**	**Output shape**
Post-cnn	SepConv2D	$3 \times 3/(3,3)$	Yes	ReLU	$512 \times 36 \times 48$	$512 \times 36 \times 48$
	Conv2D	$3 \times 3/(1,1)$	Yes	—	$512 \times 36 \times 48$	$256 \times 36 \times 48$
Upsampling-1	Upsample $\alpha = 2$	—	—	—	$256 \times 36 \times 48$	$256 \times 72 \times 96$
Skip-2	Conv2D	$1 \times 1/(1,1)$	Yes	ReLU	$256 \times 72 \times 96$	$512 \times 72 \times 96$
	Conv2D	$1 \times 1/(1,1)$	Yes	—	$512 \times 72 \times 96$	$512 \times 72 \times 96$
Upsampling-2	Conv2D	$1 \times 1/(1,1)$	Yes	ReLU	$384 \times 72 \times 96$	$768 \times 72 \times 96$
	SepConv2D	$3 \times 3/(1,1)$	Yes	ReLU	$768 \times 72 \times 96$	$768 \times 72 \times 96$
	Conv2D	$1 \times 1/(1,1)$	Yes	—	$768 \times 72 \times 96$	$128 \times 72 \times 96$
	Upsample $\beta = 2$	—	—	—	$768 \times 72 \times 96$	$128 \times 144 \times 192$
Skip-4	Conv2D	$1 \times 1/(1,1)$	Yes	ReLU	$128 \times 144 \times 192$	$256 \times 144 \times 192$
	Conv2D	$1 \times 1/(1,1)$	Yes	—	$256 \times 144 \times 192$	$64 \times 144 \times 192$
Post-Upsampling-2	Conv2D	$1 \times 1/(1,1)$	Yes	ReLU	$192 \times 144 \times 192$	$384 \times 144 \times 192$
	SepConv2D	$3 \times 3/(1,1)$	Yes	ReLU	$384 \times 144 \times 192$	$384 \times 144 \times 192$
	Conv2D	$1 \times 1/(1,1)$	Yes	—	$384 \times 144 \times 192$	$64 \times 144 \times 192$
Fusion	Conv2D	$1 \times 1/(1,1)$	—	Sigmoid	$64 \times 144 \times 192$	$1 \times 144 \times 192$
	Upsample $\gamma = 2$	—	—	—	$1 \times 144 \times 192$	$1 \times 288 \times 384$

[3] A comprehensive comparison between performance, number of parameters and execution runtime of the most recent saliency models can be found at: https://mmcheng.net/videosal/

A.2 Additional experiments

A.2.1 Additional Comparison with Recent CL methods

We further extend the performance analysis by comparing our SER strategy to other prominent CL methods in the Class-Incremental Learning setting, including recent approaches explicitly designed for Online CL, such as PEC [72] and OnPro [70]. As shown in Table ST-2, methods trained with the SER strategy (last three rows, as previously presented in Table 1 of the main paper) outperform existing methods by several percentage points, confirming the effectiveness of our approach compared to recent OCL strategies.

We also report the results obtained with TASS [38], a prior work that share some similarities with our SER method, as it introduced the use of attention maps in CL. However, these are significant differences between the two approaches. First, TASS employs a static, pre-trained saliency detector, which does not showcase the forgetting-free capabilities of saliency prediction since it is not continuously trained, unlike SER. Additionally, TASS is not designed for the OCL scenario, as it requires a large number of training epochs per task (100) and, in its original implementation, uses 50% of the classes in the first task.

Table ST-2: **Comparison with SOTA methods**, in terms of Class-IL final average accuracy (FAA).

Method	**Split Mini-ImageNet**			**Split FG-ImageNet**		
	Buffer-free methods					
PEC [72]	14.87±0.15			12.58±0.54		
TASS [38]	6.87±2.47			5.49±0.70		
	Rehearsal-based methods					
Buffer size	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**
ER [56]	14.51±5.55	16.85±2.32	19.73±1.48	10.41±0.07	6.67±0.89	10.00±1.98
A-GEM [12]	3.87±0.25	3.57±0.47	3.61±0.87	3.50±0.28	3.50±0.34	3.60±0.08
BiC [71]	7.50±1.11	9.36±0.03	9.53±1.39	4.87±0.52	4.73±1.43	4.65±0.32
FDR [5]	3.36±0.28	3.78±0.06	3.76±0.35	3.30±0.06	3.27±0.04	3.15±0.13
GEM [41]	5.45±0.14	5.92±1.10	5.76±0.51	3.17±0.10	2.59±0.13	3.43±0.04
GDumb [52]	16.14±0.48	24.12±0.96	38.67±0.04	11.95±0.16	19.29±1.74	32.19±1.51
GSS [2]	7.88±2.61	11.18±1.30	9.38±0.71	7.78±0.85	6.41±0.21	9.07±0.35
iCaRL [55]	15.64±0.13	15.81±0.53	14.58±1.58	8.97±0.66	9.32±0.03	8.84±0.76
LUCIR [25]	8.77±1.12	12.14±2.06	17.23±1.10	5.00±0.06	5.47±0.49	5.40±0.59
RPC [50]	17.14±3.77	20.08±1.09	21.00±0.46	9.96±0.99	9.32±0.71	9.29±1.12
OnPro [70]	19.34±0.26	24.29±0.67	32.23±0.51	11.73±0.30	15.63±0.13	19.95±0.66
DER++ + SER	19.13±1.62	22.92±2.25	25.35±2.56	11.71±2.36	12.97±1.62	13.73±1.95
ER-ACE + SER	27.48±2.83	33.09±1.28	35.58±1.79	20.03±3.13	23.80±2.11	28.68±0.50
CoPE + SER	26.66±2.22	33.35±4.67	45.04±2.44	18.17±2.79	27.14±1.62	34.34±3.51

A.2.2 Task-Incremental Learning setting performance

Table ST-3 reports the Task-Incremental accuracy of OCL baselines alone and when integrated with SER.

A.2.3 SER with Buffer-free methods

In Table ST-4 we report the results for both Class-Incremental and Task-Incremental settings using two common buffer-free methods: LwF [36] and oEWC [30]. Applying SER leads to performance improvements in both cases. In this case, the improvements are more evident for Task-Incremental; a marginal gain in Class-Incremental is also noticeable, though the low performance of the baseline methods limits the room for improvements.

Table ST-3: Task-Incremental accuracy of state-of-the-art methods with and without SER.

Model	**Split Mini-ImageNet**			**Split FG-ImageNet**		
Joint	63.12±1.19			56.33±2.51		
↪**SER**	**64.18**±0.60			**56.72**±1.09		
Fine-tune	34.08±2.28			28.81±1.66		
↪**SER**	**57.07**±3.44			**51.24**±2.36		
Buffer size	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**
DER++	73.07±3.07	75.11±5.61	77.71±3.04	68.65±2.14	70.24±3.97	74.74±1.14
↪**SER**	**79.75**±1.56	**82.97**±0.25	**84.10**±0.81	**72.83**±3.90	**75.40**±2.29	**78.26**±1.10
ER-ACE	71.00±3.21	75.60±3.47	77.17±4.08	66.27±0.92	69.09±3.15	70.88±5.72
↪**SER**	**77.51**±2.72	**82.22**±0.96	**83.56**±1.55	**73.08**±2.14	**75.60**±2.28	**79.46**±0.56
CoPE	68.00±0.73	71.76±2.95	74.31±2.25	63.77±2.32	67.29±3.33	69.14±2.93
↪**SER**	**72.69**±0.80	**77.57**±1.57	**84.64**±1.20	**64.79**±1.60	**73.39**±1.11	**78.66**±1.59
Dual-branch methods						
TwF	73.57±1.27	78.38±1.66	–	64.32±5.18	72.15±2.82	–
↪**SER**	**79.28**±2.24	**82.98**±0.85	–	**71.35**±1.70	**73.34**±2.94	–
DualNet	72.65±0.56	76.49±0.65	80.26±0.97	67.60±1.56	71.54±0.72	74.53±1.27
↪**SER**	**81.79**±0.59	**83.79**±0.27	**85.72**±0.40	**75.76**±0.51	**78.35**±0.36	**80.18**±0.52

Table ST-4: **Class-Incremental and Task-Incremental accuracy of non-rehearsal methods** with and without SER.

Model	**Split Mini-ImageNet**		**Split FG-ImageNet**	
	Class-IL	Task-IL	Class-IL	Task-IL
Joint	14.79±1.17	63.12±1.19	9.06±1.07	56.33±2.51
↪**SER**	**16.26**±0.30	**64.18**±0.60	**9.51**±0.93	**56.72**±1.09
Fine-tune	3.43±0.35	34.08±2.28	2.43±0.81	28.81±1.66
↪**SER**	**4.20**±0.27	**57.07**±3.44	**3.68**±0.44	**51.24**±2.36
LwF	3.18±0.41	30.61±1.80	3.25±0.45	27.55±1.64
↪**SER**	**4.22**±0.31	**48.61**±2.14	**3.57**±0.23	**36.57**±2.09
oEwC	2.68±0.24	24.10±1.55	2.38±0.23	24.98±1.15
↪**SER**	**3.08**±0.31	**35.33**±3.18	**2.55**±0.55	**26.02**±1.64

A.2.4 Effect of classification pre-training

In Table 1 of the paper we have reported the results of our experiments when the classification backbones of the baseline methods are initialized to the same weights as the saliency encoder, for a fair comparison. In this section, in order to demonstrate generalization capabilities of the SER strategy, and to ground our approach to the CL methods that exploit pre-training, we also compute performance when the classifier backbone and saliency encoder are pre-trained on a classification pre-text task (despite using classification-pretrained features appears to be in contrast to what it happens in the human brain). Differently from what described in Sec. 4.1, here we use the same disjoint subset of ImageNet classes to train the backbone of the classifier, then we initialize the saliency encoder to the same weights. Also in this setting, methods combined to SER achieve better results, as show in Table ST-5. However, the performance gain is lower than the one obtained with saliency pre-training. This is possibly due the fact that classification pre-trained features are better than saliency ones (as also evidenced by the general higher performance obtained with classification pre-training) and have reached their maximum capacity. These results confirm again the contribution of the *forgetting-free* behaviour of the saliency prediction task to classification tasks.

Table ST-5: **Class-IL and Task-IL performance when the classifier backbone and saliency encoder are pre-trained on a classification task with classes different from those available in the continual learning settings.**

Model	**Split Mini-ImageNet**			**Split FG-ImageNet**		
Buffer	**1000**	**2000**	**5000**	**1000**	**2000**	**5000**
	CLASS-IL			CLASS-IL		
DER++	30.35±0.74	30.96±0.59	32.55±1.47	15.76±0.58	16.61±0.26	16.83±0.44
↪**SER**	**31.20**±2.39	**33.91**±2.31	**37.91**±1.07	**17.06**±1.51	**20.43**±2.11	**22.53**±0.82
ER-ACE	42.33±0.57	45.84±0.50	48.77±1.28	30.91±1.02	34.09±0.57	37.49±0.47
↪**SER**	**46.56**±1.10	**50.52**±0.69	**53.23**±0.35	**32.46**±1.09	**36.08**±1.60	**40.73**±0.84
	TASK-IL			TASK-IL		
DER++	**89.98**±0.75	**91.14**±0.20	**91.37**±0.10	**83.87**±0.81	**85.61**±0.29	**86.19**±0.21
↪**SER**	89.34±0.54	90.47±0.32	91.36±0.30	82.34±0.54	84.04±0.40	84.83±0.32
ER-ACE	88.28±0.50	90.14±0.05	91.23±0.13	82.83±0.40	**85.39**±0.38	**87.29**±0.08
↪**SER**	**89.99**±0.46	**90.83**±0.20	**91.84**±0.08	**82.94**±1.15	84.25±0.95	86.51±0.25

A.2.5 Backbones Comparison

We performed other experiments including alternative backbones beyond the *classical* ResNet-18 to evaluate the generalization capability of SER across different architectures. Specifically, we applied our SER strategy with ResNet-50, MobileNet V2, and DenseNet-121. For each backbone, we compare the results obtained with the ER-ACE method with buffer size = 1000, in three scenarios: when the backbone is trained from scratch, when it is fine-tuned, and when SER is applied. As reported in Table ST-6, in all cases our SER approach leads to improved performance, thereby demonstrating its effectiveness.

A.2.6 Saliency Prediction in CL settings

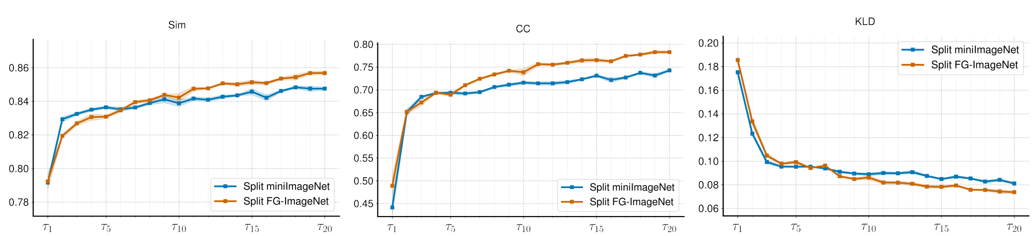

Figure SF-2: **Saliency prediction accuracy**, measured in terms of Similarity (SIM), Pearson's Correlation Coefficient (CC) and Kullback-Leibler divergence (KLD) metrics, in continual learning settings on the Split Mini-ImageNet and Split FG-ImageNet benchmarks.

Table ST-6: **Class-IL performance on ER-ACE using different backbones.**

	Split Mini-ImageNet		
Backbone	**not pre-trained**	**pre-trained**	**ER-ACE+SER**
ResNet18	15.71±0.76	20.86±3.69	27.48±2.83
ResNet50	13.38±1.41	20.34±2.72	32.16±1.23
MobileNet V2	12.76±0.54	16.55±0.69	17.77±0.41
DenseNet121	15.47±0.62	18.68±1.85	21.03±0.05

We here report quantitative performance of estimated saliency in CL settings. Fig SF-2, in particular, shows the *forgetting-free* behaviour of saliency predictions: Pearson's Correlation Coefficient (CC), Similarity (SIM) and Kullback-Leibler divergence (KLD) (metrics commonly employed for saliency predictions [10] do not degrade with the number of CL tasks.

A.2.7 Cost Analysis

Table ST-7: **Comparison of training and inference times** and parameters between SER, DualNet and TwF.

Metric	**DualNet [51]**	**TwF [8]**	**SER**
Train params	16 M	58 M	23 M
Train time	$\sim$ 6.5 h	$\sim$ 3.0 h	$\sim$ 1.0 h
Inference params	16 M	11 M	22 M
Inference time	3.45 ms	3.15 ms	7.50 ms

Table ST-8: **Training time** for the competitor methods, in their standard version, and when our SER strategy is applied.

Model	baseline	+ **SER**
LwF	$<$ 1.0 h	$\sim$ 1.5 h
oEWC	$\sim$ 1.0 h	$\sim$ 1.5 h
DER++	$\sim$ 0.5 h	$\sim$ 1.0 h
ER-ACE	$\sim$ 0.5 h	$\sim$ 1.0 h
CoPE	$\sim$ 2.0 h	$\sim$ 3.5 h

We perform cost analysis to assess the efficiency of our SER approach compared to existing methods that employ two branches, i.e., TwF [8] and DualNet [51]. It is important to note that in a continual learning settings, efficiency at training times might be more relevant than the one at inference times as the main assumption is of a deep model that keeps training from an infinite stream of data. The comparison is carried out using the ResNet18 backbone for all models. The results in Table ST-7 reveals that SER is much more efficient than DualNet and TwF at training time, while it shows higher costs at inference time (but also an accuracy gain of $\sim$10 points).
Additionally, in Table ST-8 we report the training times of the baseline version of the competitor methods, and when integrated with SER. Training time is approximately the same on both datasets, as they consist of an equal overall number of images, and the size of the buffer has a negligible impact on the training time.

A.3 Reproducibility Details

A.3.1 Hyperparameter Search

In Tables ST-9 and ST-10 we show the best hyperparameters combinations for each method.

Table ST-9: Split Mini-ImageNet

Method	Buffer	**Split-MiniImageNet**
SGD	–	lr: 0.1
LwF	–	lr: 0.01 alpha: 3.0 softmax_temp: 2.0 wd: 0.0005
oEWC	–	lr: 0.03 e_lambda: 10 gamma: 1.0
DER++	1000	lr: 0.01; alpha: 0.1; beta: 0.5;
DER++	2000	lr: 0.01; alpha: 0.1; beta: 0.5;
DER++	5000	lr: 0.01; alpha: 0.1 beta: 0.5
ER-ACE	1000	lr: 0.01; mom: 0 wd: 0
ER-ACE	2000	lr: 0.01; mom: 0 wd: 0
ER-ACE	5000	lr: 0.01; mom: 0 wd: 0
CoPE	1000	lr: 0.01; hidden_dim: 256; loss_T:0.05; p_momentum:0.9;
CoPE	2000	lr: 0.01; hidden_dim: 256; loss_T:0.05; p_momentum:0.9;
CoPE	5000	lr: 0.01; hidden_dim: 256; loss_T:0.05; p_momentum:0.9;
TwF	1000	lr: 0.01; der_alpha: 0.3; der_beta:0.9;
TwF	2000	lr: 0.01; der_alpha: 0.3; der_beta:0.9;
DualNet	1000	lr: 0.01; n_outer: 3; n_inner: 2; temp_reg = 2; alpha_reg: 10.0 slownet_beta: 0.05
DualNet	2000	lr: 0.01; n_outer: 3; n_inner: 2; temp_reg = 2; alpha_reg: 10.0 slownet_beta: 0.05
DualNet	5000	lr: 0.01; n_outer: 3; n_inner: 2; temp_reg = 2; alpha_reg: 10.0 slownet_beta: 0.05

Table ST-10: Split FG-ImageNet

Method	Buffer	**Split FG-ImageNet**
SGD	–	lr: 0.1
LwF	–	lr: 0.01 alpha: 3.0 softmax_temp: 2.0 wd: 0.0005
oEWC	–	lr: 0.03 e_lambda: 10 gamma: 1.0
DER++	1000	lr: 0.01; alpha: 0.1; beta: 0.5;
DER++	2000	lr: 0.01; alpha: 0.1; beta: 0.5;
DER++	5000	lr: 0.01; alpha: 0.1 beta: 0.5
ER-ACE	1000	lr: 0.01; mom: 0 wd: 0
ER-ACE	2000	lr: 0.01; mom: 0 wd: 0
ER-ACE	5000	lr: 0.01; mom: 0 wd: 0
CoPE	1000	lr: 0.01; hidden_dim: 256; loss_T:0.05; p_momentum:0.9;
CoPE	2000	lr: 0.01; hidden_dim: 256; loss_T:0.05; p_momentum:0.9;
CoPE	5000	lr: 0.01; hidden_dim: 256; loss_T:0.05; p_momentum:0.9;
TwF	1000	lr: 0.01; der_alpha: 0.3; der_beta:0.9;
TwF	2000	lr: 0.01; der_alpha: 0.3; der_beta:0.9;
DualNet	1000	lr: 0.01; n_outer: 3; n_inner: 2; temp_reg = 2; alpha_reg: 10.0 slownet_beta: 0.05
DualNet	2000	lr: 0.01; n_outer: 3; n_inner: 2; temp_reg = 2; alpha_reg: 10.0 slownet_beta: 0.05
DualNet	5000	lr: 0.01; n_outer: 3; n_inner: 2; temp_reg = 2; alpha_reg: 10.0 slownet_beta: 0.05

A.3.2 Task sequence details

In Tables ST-11 and ST-12 we report the combination of class order and their division into tasks employed in our experiments during the continual training. Each name corresponds to a different synset of the ImageNet dataset.

Table ST-11: Split-MiniImageNet

Task	Synsets				
τ_1	n02091244	n01770081	n03207743	n01749939	n02110063
τ_2	n02174001	n02165456	n02687172	n09246464	n02871525
τ_3	n01855672	n03062245	n04149813	n04067472	n04522168
τ_4	n02138441	n04509417	n04275548	n03888605	n01981276
τ_5	n02091831	n03400231	n02219486	n02795169	n03773504
τ_6	n03337140	n01558993	n03998194	n02129165	n03127925
τ_7	n02457408	n02108915	n04389033	n04604644	n03908618
τ_8	n02443484	n02116738	n03854065	n03544143	n09256479
τ_9	n04251144	n02606052	n02113712	n02950826	n07747607
τ_{10}	n02108551	n02108089	n07613480	n03527444	n02823428
τ_{11}	n01532829	n02981792	n02120079	n03476684	n03047690
τ_{12}	n02971356	n02074367	n06794110	n04612504	n03924679
τ_{13}	n01910747	n02105505	n03584254	n03770439	n01930112
τ_{14}	n04435653	n03347037	n03535780	n04243546	n04596742
τ_{15}	n02099601	n04418357	n02089867	n03272010	n03220513
τ_{16}	n04146614	n04443257	n02111277	n02747177	n04515003
τ_{17}	n13054560	n01843383	n07584110	n13133613	n04258138
τ_{18}	n03075370	n02966193	n03417042	n03146219	n03838899
τ_{19}	n03775546	n03017168	n03980874	n02114548	n03676483
τ_{20}	n01704323	n07697537	n02101006	n04296562	n02110341

Table ST-12: Split FG-ImageNet

Task	Synsets				
τ_1	n01943899	n01753488	n01819313	n01601694	n01695060
τ_2	n02028035	n01675722	n01498041	n01774750	n01608432
τ_3	n01685808	n01978287	n01537544	n01742172	n01924916
τ_4	n01829413	n01818515	n01494475	n01877812	n02027492
τ_5	n02058221	n01491361	n01910747	n01729977	n02018207
τ_6	n01824575	n01986214	n01860187	n01773797	n01630670
τ_7	n01796340	n01687978	n01984695	n01729322	n01833805
τ_8	n01776313	n01443537	n01560419	n02018795	n01985128
τ_9	n01677366	n01755581	n01739381	n01770081	n02013706
τ_{10}	n01978455	n02037110	n01514668	n01440764	n01855672
τ_{11}	n01756291	n01770393	n01775062	n01632458	n01820546
τ_{12}	n01496331	n01582220	n01734418	n01622779	n01632777
τ_{13}	n01806143	n01773549	n01774384	n02077923	n01740131
τ_{14}	n01484850	n01914609	n01665541	n01667778	n01847000
τ_{15}	n01667114	n01728572	n01693334	n01843383	n01950731
τ_{16}	n01514859	n02012849	n01773157	n01614925	n01795545
τ_{17}	n01944390	n02011460	n01883070	n02002556	n01798484
τ_{18}	n02051845	n01644900	n01531178	n01968897	n01698640
τ_{19}	n01592084	n01955084	n01930112	n02007558	n01735189
τ_{20}	n01751748	n01664065	n01749939	n02006656	n01828970

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: All claims made in the abstract and the introduction are demonstrated experimentally in the evaluation section.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer: [Yes]

   Justification: The paper discusses the limitations of the work in the concluding section.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory Assumptions and Proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

Answer: [NA]

Justification: The present work does not contain any theoretical result. Mathematical formulas are used for explaining the method.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental Result Reproducibility**

Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

Answer: [Yes]

Justification: The paper includes every necessary piece of information to define the model, data splits and training procedure in order to reproduce faithfully discussed results, including the number of training epochs, learning rate and all hyperparamters. The source code will be released upon acceptance.

Guidelines:

- The answer NA means that the paper does not include experiments.
- If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
- If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
- Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
- While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
  (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
  (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
  (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
  (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

   Answer: [Yes]

   Justification: The code will be released upon acceptance. The datasets used derive from ImageNet; they are already public and available online.

   Guidelines:

   - The answer NA means that paper does not include experiments requiring code.
   - Please see the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
   - The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (https://nips.cc/public/guides/CodeSubmissionPolicy) for more details.
   - The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
   - The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
   - At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
   - Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental Setting/Details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyperparameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer: [Yes]

   Justification: The details necessary to faithfully reproduce our experiments (training procedure, optimizer, number of epochs, learning rate, etc.) are included in the paper. The supplementary materials contain a list of hyperparameters for each method used in our work and any other necessary details.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment Statistical Significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer: [Yes]

   Justification: Results are provided in terms of means and standard deviations.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.

- The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
- The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
- The assumptions made should be given (e.g., Normally distributed errors).
- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments Compute Resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer: [Yes]

   Justification: The system on which the experiments were conducted is described in the paper. The execution times for main experiments are provided in Tables ST-7 and ST-8 of the supplementary materials.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code Of Ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer: [Yes]

   Justification: We reviewed and ensured that the present work respects the NeurIPS Code of Ethics at each individual part.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader Impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer: [NA]

    Justification: We propose a learning scheme to reduce forgetting regardless of the downstream model and task, thus there is no impact to the society related to the method.

Guidelines:

- The answer NA means that there is no societal impact of the work performed.
- If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
- Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.
- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer: [NA]

    Justification: We did not use any LLM and Generative models. Furthermore all datasets used for evaluation are opensource.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer: [Yes]

    Justification: References to every original owner/creator are added. Assets referenced are shown in Table CL-1.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.

Table CL-1: Assets used and licence information.

Asset	Type	License	Github / URL	Citation/Reference
Mammoth	code	MIT	aimagelab/mammoth	[9]
UNISAL	code	Apache-2.0	rdroste/unisal	[20]
Split-MiniImageNet	data	non-commercial research	yaoyao-liu/mini-imagenet-tools	[66]
Split-FG-ImageNet	data	non-commercial research	https://www.kaggle.com/datasets/ambityga/imagenet100	[58]

- The name of the license (e.g., CC-BY 4.0) should be included for each asset.
- For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
- If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
- For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.
- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New Assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer: [NA]

    Justification: This paper does not release new assets.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and Research with Human Subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer: [NA]

    Justification: This paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional Review Board (IRB) Approvals or Equivalent for Research with Human Subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

Answer: [NA]

Justification: This paper does not involve crowdsourcing nor research with human subjects.

Guidelines:

- The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.