sLoTh: An Efficient Framework for Continual Learning in Spiking Transformers

Anonymous authors
Paper under double-blind review

Abstract

Physical AI systems operate in dynamic environments where data arrives continuously, requiring models to adapt while preserving previously learned knowledge under strict memory and energy constraints. While parameter-efficient fine-tuning (PEFT) has shown promise for continual learning with vision transformers, conventional architectures rely on dense matrix multiplication and remain costly for real-world deployment. Spiking transformers provide energy-efficient event-driven computation, yet their continual learning capabilities remain largely unexplored, specifically from a PEFT perspective. We here introduce and study sLoTh, a parameter-efficient continual learning framework for pretrained spiking vision transformers. sLoTh freezes the backbone and restricts plasticity to scalable-efficient low-rank attention updates (seLoRA) and shared neuronal threshold modulation, enabling adaptation without replay buffers by updating fewer than 1% of model parameters. Experiments across CIFAR-100, Tiny-ImageNet, ImageNet-100, and ImageNet-R with up to 100 tasks demonstrate competitive rehearsal-free performance in class-incremental learning and online continual learning, while enabling $\sim 17\times$ lower inference energy consumption than conventional dense vision transformers.

1 Introduction

Continual learning is a fundamental requirement for intelligent systems deployed in changing environments, where new data must be incorporated without losing previously acquired knowledge. However, deep neural networks achieve remarkable performance on large static datasets but struggle when required to learn continually. In such settings, as described in Figure 2, models suffer from ***catastrophic forgetting*** Van de Ven and Tolias (2019); Kirkpatrick et al. (2017), where learning new tasks overwrites previously acquired knowledge due to unconstrained plasticity. This challenge is aggravated in resource-constrained environments where storing past observations for replay or repeatedly updating a large model may be impractical. Recent years have seen a strong shift from CNNs to vision transformers Dosovitskiy et al. (2020) for downstream tasks due to their transferable pretrained representations. However, naively fine-tuning pretrained transformers for continual learning leads to severe forgetting and incurs substantial computational and memory overhead Wu et al. (2025); Zhou et al. (2024b). Parameter-Efficient Fine-Tuning (PEFT) methods, including prompt tuning Wang et al. (2022b;a), adapters Li et al. (2025), and low-rank adaptation (LoRA) Hu et al. (2022), address this problem partly by updating only a small subset of parameters while freezing the pretrained backbone. These approaches substantially reduce the number of trainable parameters and the associated adaptation cost while maintaining strong continual learning performance Zhang et al. (2023); Liang et al. (2025).

However, two challenges remain for continual adaptation under stringent resource constraints. First, standard artificial neural network (ANN) based vision transformers rely on dense matrix multiplication, resulting in high energy consumption. This energy efficiency is demonstrated at the hardware level by neuromorphic processors such as Darwin Shen et al. (2016); Ma et al. (2023), Loihi Davies et al. (2018), DYNAP-SE Moradi et al. (2017), DYNAP-SE2 Richter et al. (2024), which exploit event-driven, spike-based computation to enable low-power neural processing. Second, PEFT-based continual learning has been studied almost exclusively in ANNs Wu et al. (2025); Wang et al. (2022b;a); Smith et al. (2023); He et al. (2025), while the interaction between parameter-efficient adaptation and *spiking neural networks*, known for their extreme energy-efficient compu-

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

tation, remains insufficiently studied. Spiking transformers such as SpikFormer Zhou et al. (2023), QKFormer Zhou et al. (2024a), and SpikingFormer Zhou et al. (2026) replace dense matrix multiplications with spike-based computation, significantly improving energy efficiency. This inspires us to study whether pretrained spiking transformers can support exemplar-free continual learning across their variants while updating only a small fraction of model parameters. We believe this is particularly relevant to physical AI and edge intelligence, where vision models need to adapt after deployment while operating in resource-constrained environments, such as with limited battery life.

In this work, we introduce **sLoTh**, a rehearsal-free continual learning framework for spiking vision transformers for such resource-constrained environments. The proposed framework combines three mechanisms: **seLoRA**, an efficient low-rank attention adaptation module that enables efficient representational updates; **threshold modulation**, which dynamically adjusts neuronal firing thresholds to regulate intrinsic plasticity in spiking networks; and **multi-objective distillation and regularisation** that preserves previously learned knowledge without requiring data replay. Our approach enables continual adaptation without network expansion Han et al. (2023) or the use of exemplar buffers Ni et al. (2025); Buzzega et al. (2020); Liu et al. (2025). Accordingly, we evaluate our method across the ***Task Incremental Learning*** (TIL), ***Class Incremental Learning*** (CIL), and ***Online Continual Learning*** (OCL) settings, with the necessary algorithmic modifications. Our contributions in this work are: (1) We study parameter-efficient continual learning of spiking vision transformers, focusing specifically on efficient exemplar-free continual learning. (2) We introduce **sLoTh**, a data rehearsal-free continual learning framework that combines shared **threshold modulation** with efficient low-rank attention adaptation (**seLoRA**) to enable stable adaptation without replay buffers or architectural expansion. (3) Unlike prior LoRA-based continual learning approaches Wu et al. (2025); Liang and Li (2024); He et al. (2025) that rely on task-specific modules which **scales linearly w.r.t resource requirements** with increasing number of tasks, sLoTh performs **shared adaptation** optimised through a **multi-objective regularisation** while preserving intrinsic plasticity in spiking networks. (4) We systematically study how these plasticity mechanisms interact across continual-learning regimes. We observe that joint threshold and seLoRA adaptation is effective in multi-epoch CIL/TIL settings, while individual mechanisms provide better adaptation for OCL settings. We further investigate task-free novelty detection by developing loss-based and representation-based triggers.(5) Our approach updates **less than 1% of model parameters** while leveraging the efficiency of spiking transformers, which achieve $\sim$**17**$\times$ lower energy consumption than conventional dense vision transformer architectures.

2 Related Work

2.1 Continual Learning

Continual learning (CL) enables models to learn from sequentially arriving data while mitigating catastrophic forgetting. Existing approaches are commonly grouped into regularisation-based, rehearsal-based, and architecture-based methods Van de Ven and Tolias (2019). Regularisation-based approaches constrain parameter updates through importance-weighted penalties such as EWC Kirkpatrick et al. (2017), R-EWC Liu et al. (2018), and Synaptic Intelligence Zenke et al. (2017). Rehearsal-based methods approximate the joint training distribution by replaying stored samples or features, including Experience Replay (ER) Rolnick et al. (2019), GEM Lopez-Paz and Ranzato (2017), A-GEM Chaudhry et al. (2018b), and Dark Experience Replay (DER) Buzzega et al. (2020). For OCL, replay-based strategies remain dominant Rolnick et al. (2019); Buzzega et al. (2020). Architecture-based approaches expand or gate model components to isolate task-specific parameters, though most were developed for convolutional networks and do not directly address parameter-efficient adaptation of pretrained transformers.

2.2 Parameter-Efficient Fine-Tuning for Continual Adaptation

Parameter-Efficient Fine-Tuning (PEFT) enables adaptation of large pretrained transformers by updating only a small subset of parameters while keeping the backbone frozen Zhou et al. (2024b). Existing methods include prompt-based approaches such as L2P Wang et al. (2022b), Coda-Prompt Smith et al. (2023), and DualPrompt Wang et al. (2022a); adapter-based modules inserted within transformer layers; and low-rank adaptation methods such as LoRA and its continual variants, including SD-LoRA Wu et al. (2025) and Online LoRA Wei et al. (2025).

2.3 Continual Learning in Spiking Networks

Spiking Neural Networks (SNNs) enable event-driven computation through membrane potential accumulation and threshold-based firing, offering improved energy efficiency compared to conventional neural networks. Recent work extends transformer architectures into this domain through spike-based attention mechanisms, as demonstrated by models such as Spikformer Zhou et al. (2023), SpikingFormer Zhou et al. (2026), and QKFormer Zhou et al. (2024a). However, these studies don't focus on continual learning. Continual learning in spiking networks typically relies on CNN or MLP backbones Han et al. (2023); Shen et al. (2024), sometimes with rehearsal buffers Ni et al. (2025) or via task-specific dynamic thresholds for Spikformer Zhou et al. (2023) trained from scratch Nagabhushana et al. (2026). However, no prior works analyse parameter-efficient fine-tuning (PEFT) methods for continual learning using pretrained spiking transformers.

3 Methodology

We propose a rehearsal-free continual learning framework for spiking transformers, designed for energy-efficient adaptation in resource-constrained environments. By restricting plasticity to lightweight adaptation modules while keeping the pretrained backbone frozen, the proposed approach enables continual adaptation with minimal parameter updates and computational overhead. Our primary focus is on the Class-Incremental Learning (CIL) and Task-Incremental Learning (TIL) settings. We additionally extend the framework to the more challenging Online Continual Learning (OCL) setting with minor modifications, enabling adaptation in a single-pass streaming setting without explicit task boundaries.

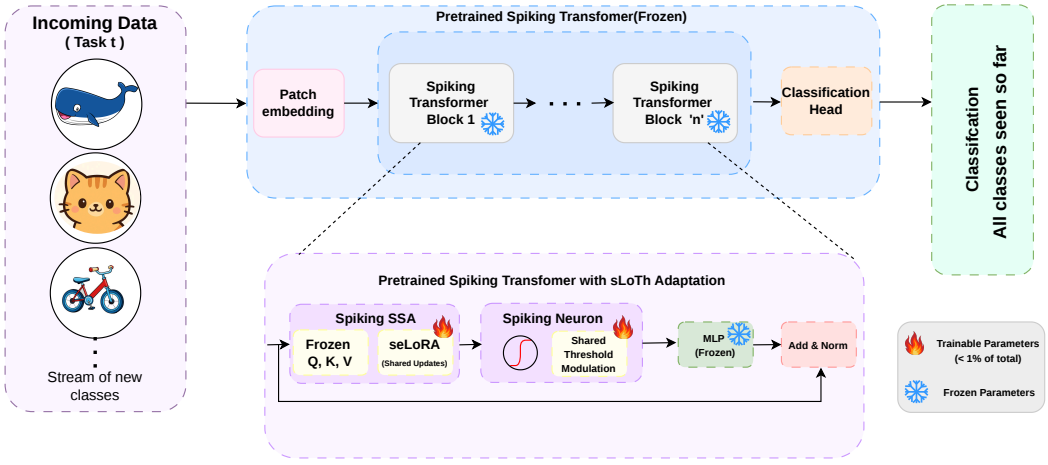

Figure 1: Overview of the sLoTh framework for parameter-efficient adaptation of a pretrained spiking transformer.

3.1 Problem Formulation

We consider a sequence of tasks $\{\mathcal{D}_t\}_{t=1}^{T}$, where each task $\mathcal{D}_t = \{(x_i, y_i)\}_{i=1}^{N_t}$ introduces a disjoint label space $\mathcal{C}_t$. The cumulative label space after task t is $\mathcal{C}_{\leq t} = \bigcup_{k=1}^{t} \mathcal{C}_k$. We evaluate **Full sLoTh**, comprising seLoRA and threshold modulation, under three continual learning protocols i.e CIL, TIL and OCL. Refer to appendix A.1 for detailed problem formulation.

3.2 Model Plasticity

Let $f_\theta(\cdot)$ denote a pretrained spiking transformer with frozen backbone parameters θ. To avoid catastrophic forgetting caused by representation drift, we restrict plasticity to lightweight adaptation parameters while keeping the backbone fixed. Low-rank adaptation (LoRA) provides a parameter-efficient strategy for continual learning Hu et al. (2022); Liang and Li (2024); Wu et al. (2025);

Wei et al. (2025), but many approaches introduce task-specific modules Li et al. (2025); Liang et al. (2025) leading to parameter growth and feature drift in continual learning. To address this, we introduce *channel-wise excitability modulation*, which adjusts neuronal firing thresholds instead of synaptic weights to adapt feature responses while preserving the pretrained representation. For multi-epoch settings such as CIL and TIL, we additionally enable scalable, efficient low-rank attention updates to increase adaptation capacity.

Adaptation Recipes. We jointly optimize threshold modulation and seLoRA ($\psi = \{\delta, \phi\}$) for CIL and TIL. For OCL, we evaluate the joint formulation (Full sLoTh) together with threshold-only ($\psi = \{\delta\}$) and seLoRA-only ($\psi = \{\phi\}$) variants.

$$\psi = \begin{cases} \{\phi, \delta\} & \text{CIL/TIL,} \\ \{\phi, \delta\}, \{\delta\}, \text{ or } \{\phi\} & \text{OCL.} \end{cases} \tag{1}$$

3.2.1 Channel-wise Excitability Modulation

In spiking networks, neurons emit spikes when their membrane potential exceeds a firing threshold. Modulating this threshold alters neuronal excitability without changing synaptic weights, a mechanism widely observed in biological neural systems where neuromodulators regulate neuronal gain and firing thresholds Hammouamri et al. (2022); Ding et al. (2022).

We introduce learnable *per-channel* threshold offsets that modulate neuronal excitability:

$$v_{\text{th}}(c) = v_{\text{th}}^{\text{base}} + \delta_c, \tag{2}$$

where $v_{\text{th}}^{\text{base}}$ is the pretrained base threshold and δ_c is learnable offsets for each feature channel. Each parameter δ_c controls the excitability of an entire feature map across spatial locations, allowing the model to selectively amplify or suppress specific channels in response to new tasks. Unlike weight-space adaptation methods such as LoRA Hu et al. (2022), threshold modulation adjusts intrinsic neuronal excitability rather than synaptic connectivity. Because this mechanism modulates neuronal excitability without directly altering pretrained feature projections, it introduces less representational disruption than weight-space adaptation methods. This behaviour is particularly beneficial for continual learning approaches that rely on stable feature-space prototypes.

3.2.2 Low-Rank Attention Modulation

We incorporate low-rank attention modulation inspired by LoRA Hu et al. (2022). The low matrices A and B form the learnable parameters $\phi_i = \{A_i, B_i\}_i$. In our design, we refer to this adaptation scheme as **seLoRA** (*Scalable-Efficient LoRA*), where a single shared low-rank module provides parameter-efficient plasticity without allocating task-specific adapters and is optimised via a stabilised training objective (see 3.3). To motivate the choice of lightweight plasticity mechanisms, we first evaluate them in a conventional fine-tuning setting, without continual learning. As reported in Table 9, low-rank adaptation and threshold modulation provide complementary parameter-efficient adaptation capabilities. These results *motivate retaining both weight-space adaptation through seLoRA* and *intrinsic neuronal adaptation through threshold modulation* as candidate sources of plasticity in the continual learning setting.

Several continual learning methods employ LoRA-style adapters that expand with the number of tasks or rely on explicit task boundaries Wei et al. (2025); Wu et al. (2025); Liang et al. (2025). Our seLoRA instead uses a single shared low-rank module across the task sequence, avoiding task-specific parameter growth. We evaluate seLoRA jointly with threshold modulation in CIL and TIL, and separately in OCL as described by the adaptation recipes 3.2.

3.3 Stabilized Training Objective

The training objective balances *adaptation* to new classes with *stabilisation* of previously learned representations:

$$\mathcal{L} = \mathcal{L}_{\text{CE}} + \lambda_{\text{KL}} \mathcal{L}_{\text{KL}} + \lambda_{\text{local}} \mathcal{L}_{\text{local}} + \lambda_{\text{global}} \mathcal{L}_{\text{global}} \tag{3}$$

where $\mathcal{L}_{\text{CE}}$ refers to the classifier trained using cross-entropy over the current task. The stability term comprises teacher-student logit distillation and local and global parameter anchors. Further details are provided in Appendix A.3 and analysis in section A.7.

3.4 Classifier and Prototype Inference

Given a feature representation $z = f_{\theta,\psi}(x) \in \mathbb{R}^d$, logits are computed using a cosine classifier $\ell_c(x) = s \cdot \frac{z}{|z|_2} \cdot \frac{w_c}{|w_c|_2}$ where $w_c \in \mathbb{R}^d$ denotes the classifier weight for class c and s is a fixed temperature. After completing task t, the classifier columns corresponding to classes $w_{c \in \mathcal{C}_t}$ are frozen, and only the weights of newly introduced classes are updated in subsequent tasks. While the cosine classifier is used during training, we employ Nearest Class Mean (NCM) classification at inference time to reduce recency bias. This method is widely used in the literature and has been employed in prior work, including Rebuffi et al. (2017); Wang et al. (2022b;a); Hou et al. (2019). After task t, a prototype for each class is computed as μ_c. During inference, predictions are obtained by nearest prototype matching as $\hat{y}$.

$$\mu_c = \frac{1}{|i : y_i = c|} \sum_{y_i = c} f_{\theta,\psi}(x_i). \quad and \;\; \hat{y} = \arg\max_{c \in \mathcal{C}_{\leq t}} \cos(f_{\theta,\psi}(x),\, \mu_c)\,. \tag{4}$$

3.5 Online Novelty Detection in Event-Based (Spiking) Transformers

Training loss based: In the OCL setting, task boundaries are not provided to the model Aljundi et al. (2019b); Ye and Bors (2024). The data, therefore, arrive as a continuous stream, and distribution shifts may indicate the introduction of previously unseen classes. Detecting such shifts is closely related to concept drift detection in streaming learning systems Yang et al. (2024). We detect distribution changes online using the dynamics of the training loss. Specifically, we maintain an exponential moving average (EMA) of the cross-entropy loss by $\bar{\mathcal{L}}_b = (1-\alpha)\bar{\mathcal{L}}_{b-1} + \alpha\mathcal{L}_b$ where $\mathcal{L}_b$ is the loss at batch b and α controls the smoothing factor. At each step, we compute the loss ratio $r_b = \mathcal{L}_b / \bar{\mathcal{L}}_b$. A novelty trigger fires when $r_b > \gamma$, where γ is a predefined threshold hyperparameter (e.g., $\gamma = 2$ indicates that the loss has doubled relative to the running average). Such abrupt increases in loss may indicate distribution shifts or the appearance of previously unseen classes Yang et al. (2024).

We use this mechanism as a lightweight task-free trigger for online adaptation, prioritizing simplicity and low memory overhead over the robustness of dedicated concept-drift detection methods. Unlike several **ANN-based OCL** methods that rely on replay buffers, clustering, or explicit task-boundary supervision, our approach prioritizes minimal memory overhead and computational simplicity, making it suitable for resource-constrained streaming settings. When a trigger is activated, the model snapshots the current adaptation parameters ψ as a new teacher for the local anchor and registers newly observed classes. More advanced distribution-shift detection mechanisms may further improve robustness in future work.

Representation Thresholding (RT): To further evaluate the proposed method on a more robust novelty detector, we design an additional variant that performs representation-based (RT) novelty detection using intermediate features from the pretrained spiking transformer. Rather than relying on the loss-based trigger, we extract intermediate representations De La Jara et al. (2025) from frozen early transformer blocks (e.g., 4 and 6). For each class, we maintain only 10 prototype representations, introducing negligible memory overhead compared to replay buffers or auxiliary networks. Incoming samples are compared against these prototypes to detect distribution shifts, after which Threshold Modulation/seLoRA adapts only the later blocks (7–10), preserving stable pretrained representations while enabling efficient online adaptation. This design separates novelty detection from adaptation: the frozen early blocks provide a stable feature space for identifying samples that deviate from previously observed class representations, while the later blocks remain available for plastic adaptation. The prototype-based criterion, therefore, provides a complementary alternative to loss-based triggering, allowing novelty to be assessed directly from the learned representation rather than indirectly through optimization loss. By restricting adaptation to the later blocks, RT further limits the impact of online updates on the representations used for novelty detection.

3.6 Extension to Online Continual Learning

For OCL, data arrive in a single-pass stream without an explicit task boundaries. We therefore replace task-based teacher updates with the novelty-detection mechanism described in Section 3.5, which triggers adaptive teacher snapshotting in response to detected distribution shifts.

4 Experiments

We evaluate sLoTh, a parameter-efficient continual learning framework designed for spiking transformers. Our experiments address two key questions - (1)*Which PEFT mechanisms are compatible with spiking transformers under resource constraints?* (2) *Can rehearsal-free event-based adaptation compete with replay-based continual learning methods, and how is it connected to task types such as CIL and OCL?*

4.1 Experimental Setup

We primarily use QKFormer Zhou et al. (2024a) as the pretrained spiking-transformer backbone. To assess the transferability of sLoTh across transformer architectures, we additionally evaluate it on MaxFormer Fang et al. (2026), SpikingFormer Zhou et al. (2026), and SDT-v3 Yao et al. (2025). For comparison with existing PEFT methods, we also report results using a pretrained ViT-B/16 Dosovitskiy et al. (2020) backbone, where applicable, to assess transferability across different transformer architectures. These results suggest that the proposed framework generalizes with minor architectural modifications and limited hyperparameter tuning. We report two standard continual learning metrics: **Average Accuracy (AA)** and **Average Forgetting (F)**. Let $a_{t,i}$ denote the accuracy on task i after learning task t, with T total tasks. The final average accuracy is defined as $\text{AA} = \frac{1}{T}\sum_{i=1}^{T} a_{T,i}$, while forgetting task i is $F_i = \max_{t\in\{1,\dots,T\}}\{a_{t,i} - a_{T,i}\}$. The average forgetting is then $F = \frac{1}{T-1}\sum_{i=1}^{T-1} F_i$. AA measures retained performance across tasks, whereas forgetting quantifies the degradation of stability during continual adaptation.

4.2 Class Incremental Learning

We evaluate sLoTh on spiking transformers, which offer lower energy consumption than ANN-based transformers (Table 5). In Table 1, several ViT-based PEFT methods perform strongly under coarse task splits, but their performance degrades or becomes computationally infeasible as the number of tasks increases. In contrast, Full sLoTh scales favourably with finer task granularity, improving from **81.23%** for 10 tasks to **84.65%** and **87.22%** for 20 and 50 tasks, respectively, and outperforming the compared methods at the finer splits. Further energy and parameter comparisons are provided in Table 6. We next evaluate sLoTh on the higher-resolution **ImageNet-R** dataset.

Table 1: CIFAR100 CIL comparison between pretrained ViT with PEFT Continual Learning methods v/s pretrained QKFormer on **Full sLoTh** for different task granularity, i.e., T represents the number of tasks. Parameters are in millions. **First** and Second best models are highlighted.

Methods	T=10	T=20	T=50	Total Param(M)	#updates(M)	Wall clock
L2P Wang et al. (2022b)	83.18±1.20	79.51±0.67	67.95±2.12	172	**0.12**	0.366
DualPrompt Wang et al. (2022a)	81.48±0.86	80.44±1.38	72.5±1.08	172	0.86	0.343
CODA-Prompt Smith et al. (2023)	86.31±0.12	81.36±0.88	73.77±0.98	172	4.6	0.395
InfLoRA Liang and Li (2024)	86.75±0.35	80.97±0.74	70.68±1.26	172	0.51	0.367
SD-LoRA Wu et al. (2025)	**88.01±0.31**	OOM	OOM	172	0.39	0.498
Full sLoTh	81.23±1.74	**84.65±1.46**	**87.22±1.59**	**64.32**	0.44	0.239

From Table 3, we observe that several ViT-based PEFT approaches achieve strong performance under coarse task splits, but their accuracy degrades as the number of incremental tasks increases. For example, SD-LoRA achieves **79.15%** accuracy for $T = 5$ but drops to **75.26%** for $T = 20$, indicating increased sensitivity to multiple incremental updates. In contrast, Full sLoTh exhibits a more stable scaling behaviour as the number of tasks increases. Although its accuracy is lower under the coarse $T = 5$ split (**71.61%**), it improves to **77.10%** at $T = 20$, outperforming the compared methods under the finer task partition. Although SD-LoRA provides competitive performance, its task-specific low-rank matrices are stored and the direction vectors are updated separately for

each task, resulting in parameter and memory growth that scales linearly with the number of tasks. This motivates our use of shared low-rank adaptation in sLoTh, which avoids task-specific parameter accumulation while retaining parameter-efficient adaptation. This suggests that the proposed lightweight adaptation remains effective when the model is exposed to more incremental updates, rather than relying primarily on coarse task-level adaptation. Importantly, this scalability is achieved while retaining the memory and energy-efficiency advantages of the spiking architecture and updating only a small fraction of the model parameters. Overall, the results indicate that Full sLoTh provides a better combination of continual adaptation and computational efficiency, particularly as the incremental task granularity increases. On MaxFormer, Full sLoTh achieves **76.52%**, **77.90%**, and **79.58%** for $T = 5$, 10, and 20, respectively, demonstrating consistent performance as the task granularity increases.

We further evaluate the proposed framework on *event-based* benchmarks such as N-Caltech101 and DVS-Gesture, under the class-incremental learning setting. We compare against representative state-of-the-art PEFT continual learning methods (CODA-Prompt and SD-LoRA) in MaxFormer (Fang et al., 2026). N-Caltech101 is evaluated using 10 tasks (Task-1 contains 11 classes, followed by 10 classes per task), while DVS-Gesture is evaluated using 5 tasks (2 classes per task).

Table 2: Performance comparison on event-based benchmarks (N-Caltech101, $T = 10$; DVS-Gesture, $T = 5$).

Method	**N-C101**	**DVS-G**
CODA-Prompt	73.87	86.38
SD-LoRA	75.10	**89.55**
Full SLoTh	**76.21**	88.91

Table 3: CIL under domain shift on ImageNet-R. T denotes the number of task splits. Accuracy is reported as mean ± std over three runs. **First** and Second best models are highlighted.

Backbone	**Method**	***T = 5***	***T = 10***	***T = 20***	**Params**
QKFormer	Full Fine tuning	64.92 ± 0.87	60.57 ± 1.06	49.95 ± 1.31	64.32M
QKFormer	L2P Wang et al. (2022b)	73.04 ± 0.71	71.26 ± 0.44	68.97 ± 0.51	64.32M
QKFormer	DualPrompt Wang et al. (2022a)	69.99 ± 0.57	68.22 ± 0.20	65.23 ± 0.45	64.32M
QKFormer	CODA-Prompt Smith et al. (2023)	76.63 ± 0.27	74.05 ± 0.41	69.38 ± 0.33	64.32M
QKFormer	InfLoRA Liang and Li (2024)	76.95 ± 0.23	74.75 ± 0.64	69.89 ± 0.56	64.32M
QKFormer	SD-LoRA Wu et al. (2025)	**79.15 ± 0.20**	77.34 ± 0.35	75.26 ± 0.37	64.32M
QKFormer	**Full sLoTh**	71.61 ± 0.92	76.17 ± 1.71	77.10 ± 1.13	64.32M
MaxFormer	**Full sLoTh**	76.52 ± 1.42	**77.9 ± 1.37**	**79.58 ± 2.05**	63.99M
SpikingFormer	**Full sLoTh**	68.18 ± 1.7	71.85 ± 1.01	75.29 ± 1.94	66.34M
SDT-v3	**Full sLoTh**	70.39 ± 0.93	73.88 ± 1.61	75.29 ± 0.78	19M
ViT_{base}	**seLoRA**	72.54 ± 1.28	75.29 ± 1.15	77.89 ± 2.53	86M

4.3 Extension to Online Continual Learning

We further evaluate sLoTh on the Online Continual Learning (OCL) setting, where data arrive in a single-pass stream, and explicit task boundaries are unavailable during training. Unlike many existing OCL approaches, sLoTh does not rely on replay buffers or stored training samples. Instead, the framework maintains only lightweight per-class prototype statistics, substantially reducing memory overhead while preserving representative information for decision boundaries in streaming environments. Consistent with the CIL setting, we focus here on energy-efficient OCL using spiking transformers. Table 4 compares sLoTh against several replay-based OCL methods on CIFAR-100 ($T = 10$) and Tiny-ImageNet ($T = 100$). Unlike these baselines, which rely on replay buffers containing up to **5k** and **10k** stored samples, respectively, sLoTh operates without replay memory and maintains only lightweight prototype statistics. Despite these constraints, Full sLoTh achieves **55.12%** accuracy with **12.79** forgetting on CIFAR-100, closely matching the replay-based MOE-MOSE baseline (**55.62%**). These results suggest that structured intrinsic plasticity together with lightweight attention modulation can effectively mitigate catastrophic forgetting even under strict streaming conditions. The effectiveness of sLoTh becomes more evident on **Tiny-ImageNet** with **100 tasks**. Full sLoTh achieves **42.90%** accuracy, outperforming the evaluated replay-based methods, including OCM (**31.94%**), OnPro (**26.38%**), and MOE-MOSE (**38.41%**). The individual variants achieve higher accuracy in strict single-pass streaming, with Threshold-only reaching **61.41%** and seLoRA-only **65.12%**. Figure 3 compares task-aware and boundary-free online learning. With

known task boundaries, Threshold-only achieves **95.7%**, while removing boundaries reduces it to **62.1%**; seLoRA-only similarly decreases from **85.4%** to **65.7%**. This highlights the difficulty of adapting to boundary-free streaming. Threshold-only also requires at least **10**$\times$ fewer parameter updates than seLoRA-based variants.

Table 4: Online Continual Learning comparison under single-pass streaming. Replay-based results are taken from S6MOD under the largest buffer sizes (CIFAR100: M=5k, Tiny-ImageNet: M=10k). **sLoTh operates without any replay buffer** has only per-class mean prototype. Accuracy is reported as mean $\pm$ std over 3 runs. **First** and Second best models are highlighted.

Backbone	Method	CIFAR100 ($T = 10$)		Tiny-ImageNet ($T = 100$)	
		Avg Acc	Forgetting	Avg Acc	Forgetting
ViT	ER Rolnick et al. (2019)	39.41±1.81	13.29	24.71±2.52	40.77
	OCM Guo et al. (2022)	42.22±1.06	**3.76**	31.94±1.19	15.92
	OnPro Wei et al. (2023)	41.59±1.38	6.72	26.38±2.18	20.32
	OCM-CCLDC Wang et al. (2024)	51.43±1.37	3.99	39.25±0.88	15.56
	OnPro-CCLDC Wang et al. (2024)	50.01±0.85	10.55	38.18±1.02	16.17
	MOSE Yan et al. (2024)	54.53±0.78	13.60	38.71±0.44	15.51
	MOE-MOSE Yan et al. (2024)	55.62±0.72	12.81	38.41±0.53	13.94
QKFormer	Full sLoTh	55.12 ± 1.18	12.79	42.90 ± 2.56	14.63
	sLoTh only with Thresholds	60.25 ± 0.98	11.36	**61.41 ± 1.05**	**12.32**
	sLoTh only with seLoRA	**63.88 ± 1.03**	13.85	**65.12 ± 2.52**	15.16
MaxFormer	Full sLoTh + RT	39.60 ± 2.07	13.88	46.28±1.70	11.08
	sLoTh (Thresh) + RT	59.33 ± 1.29	12.52	60.47±2.38	15.8
	sLoTh (seLoRA) + RT	56.81 ± 1.55	11.76	62.±1.87	12.35

To further assess the generality and efficiency of sLoTh under Online Continual Learning, we evaluate Split-CIFAR-100 with 10 tasks using a spiking QKFormer backbone with the standard loss-based novelty detector. We additionally evaluate a **representation-thresholding (RT)** variant using a MaxFormer backbone, where distribution shifts are detected from intermediate representations of frozen early transformer blocks using a small set of class prototypes. Once a shift is detected, RT restricts adaptation to later blocks, preserving the early pretrained representation while enabling lightweight online adaptation. This comparison evaluates representation-space novelty detection as an alternative to the standard loss-based trigger. Overall, these results demonstrate that the **sLoTh framework can match or surpass replay-based continual learning methods while remaining energy- and memory-efficient.** Similarly, from Table 12 we observe that the Full sLoTh method outperforms prior state-of-the-art methods with ViT backbone on ImageNet-100 with 50 tasks by at least 1.76%. Online-LoRA Wei et al. (2025) when transferred to pretrained **QKFormer** backbone Zhou et al. (2024a) gives us 61.87, which is approximately equivalent to Full sLoTh's performance. Moreover, seLoRA is also applicable to ViT (55.68%), and for such large task counts, it achieves only 70% performance.

4.4 Plasticity Across Continual Learning Regimes

How is parameter-efficient plasticity of the spiking transformer connected to the learning regime? We here sought to answer this question. Table 9 indicates PEFT in spiking networks can be mediated by seLoRA only, Threshold only or both (Full sLoTh) with varying classification accuracies. From Table 6 we observe that, under CIL, where multiple epochs are available per task, the interaction between scalable-efficient low-rank modulation (seLoRA) and threshold adaptation becomes beneficial. seLoRA provides weight updates, while threshold modulation regulates neuronal excitability as new classes are introduced. Their combination significantly improves performance, achieving **81.23%** accuracy for Full sLoTh. From Table 6, we also observe that for CIL, seLoRA only and Threshold only can also independently attain nearby performances of 72.63% and 69.39%, respectively. Whereas, during OCL, different variants (seLoRA only or Threshold only) yield higher immediate accuracy (**63.88%** and **60.25%** vs. **55.12%** for Full sLoTh). This behaviour arises because the OCL setting observes each sample only once, favouring highly plastic adaptation mechanisms that can rapidly adjust model parameters. Here, seLoRA parameters are randomly initialised and must learn meaningful projection subspaces. In OCL settings, the additional constraints introduced by the full objective (Full sLoTh) can slow this adaptation, leading to slightly lower short-term accu-

Table 5: Online continual learning results on Split-CIFAR-100 with 10 tasks. Prior methods use a ViT-B/16 Dosovitskiy et al. (2020) backbone with 86M parameters, while sLoTh uses the $\tilde{6}4$M-parameter QKFormer and MaxFormer backbone. We report average accuracy, forgetting, the fraction of trainable parameter updates, and inference energy. Energy values marked with † and ⋆ are taken from Table 2 of Zhou et al. (2024a) and from Fang et al. (2026) respectively, for energy measurements.**First** and Second best models are highlighted.

Methods	Avg Acc (↑)	Forgetting (↓)	# Updates	Energy
AGEM Chaudhry et al. (2018b)	12.67 ± 1.87	82.51 ± 2.27	Full	254.84
ER Rolnick et al. (2019)	44.85 ± 1.83	44.67 ± 4.29	Full	254.84
EWC++ Chaudhry et al. (2018a)	10.61 ± 0.74	84.10 ± 1.11	Full	254.84
MIR Aljundi et al. (2019a)	48.36 ± 3.11	43.41 ± 1.02	Full	254.84
PCR Lin et al. (2023)	48.48 ± 0.15	46.23 ± 1.29	Full	254.84
LODE (DER++) Liang and Li (2023)	44.29 ± 1.48	45.54 ± 3.32	Full	254.84
EMA (DER++) Soutif-Cormerais et al. (2023)	42.28 ± 4.36	55.59 ± 1.48	Full	254.84
EMA (RAR) Soutif-Cormerais et al. (2023)	47.10 ± 0.82	50.01 ± 0.35	Full	254.84
Online-Lora Wei et al. (2025)	49.40 ± 1.36	41.74 ± 2.58	0.623	254.84
Full sLoTh	55.12 ± 1.18	12.79 ± 2.28	0.439	**38.91**†
sLoTh only thresholds	60.25 ± 0.98	**11.36 ± 2.41**	**0.037**	**38.91**
sLoTh only seLoRA	**63.88 ± 1.03**	13.85 ± 3.97	0.402	**38.91**
Full sLoTh + RT	39.60 ± 2.07	13.88±1.27	0.321	**14.87**⋆
sLoTh (Thresh) + RT	59.33 ± 1.29	12.52±1.04	0.063	14.87
sLoTh (seLoRA) + RT	56.81 ± 1.55	11.76±2.6	0.261	14.87

racy. These results indicate that, while seLoRA-only or Threshold-only is beneficial for immediate adaptation in streaming settings, dual plasticity (Full sLoTh) becomes essential when learning must remain consistent across longer task sequences.

Table 6: Component interaction under CIL and OCL (CIFAR-100, 10 tasks).

Variant	CIL Avg Acc	OCL Avg Acc (loss based)	OCL Avg Acc(RT)	Param updates
seLoRA only (w/o Threshold)	72.63±1.24	**63.88±1.03**	56.81±1.55	0.402
Threshold only (w/o seLoRA)	69.39±1.01	60.25±0.98	59.33±1.29	$\mathbf{0.037}_{(\mathbf{10\times\downarrow})}$
Full sLoTh (seLoRA + Threshold)	**81.23±1.74**	55.12±1.18	39.±2.07	0.439

Threshold Modulation for Parameter Efficient Continual Learning: Spiking thresholds are a unique attribute of spiking neural networks. From Table 9 we observe that fine-tuning only thresholds reaches an accuracy of 70.56% compared to 80.89% with LoRA. From Tables 12 and 6, we notice that for the continual learning scenario, threshold modulation is an effective parameter-efficient fine-tuning method (requiring atleast $10\times$ lower parameter updates than LoRA). The effect is most pronounced in the OCL scenario, where Threshold achieves 60.25% and 66.91% accuracy on CIFAR-100 and ImageNet-100, respectively, while requiring substantially fewer parameter updates than seLoRA.

5 Discussion

We propose **sLoTh** as a framework for studying parameter-efficient plasticity in spiking transformers across CIL, TIL, and OCL. Our results show that threshold modulation provides highly parameter-efficient plasticity, while seLoRA provides additional adaptation capacity in multi-epoch settings. Joint adaptation is effective in CIL and TIL, whereas individual plasticity mechanisms achieve higher accuracy in the evaluated OCL settings, indicating that plasticity does not transfer directly across continual-learning regimes. The component analysis in Table 7 further shows that removing any of the regularisation terms substantially degrades performance, with KL regularisation having the largest effect. The component interaction results in Table 6 further show that joint threshold and seLoRA adaptation is effective inCIL, whereas individual plasticity mechanisms achieve higher accuracy in the evaluated OCL setting on CIFAR-100 with 10 tasks. Despite operating without replay memory, sLoTh also outperforms the evaluated replay-based methods on the reported Tiny-ImageNet OCL benchmark, demonstrating the potential of lightweight intrinsic and attention-based plasticity for memory-constrained continual learning.

AI USE STATEMENT

In this work, we used generative AI tools to write and polish the text. We have not used generative AI tools for other tasks which requires any disclosure such as- Generate synthetic data sets, help develop theoretical models or conceptual frameworks, formulate mathematical claims, provide critical ingredients for proving mathematical claims, assist in the writing of proofs, propose or refine hypotheses, design or provide feedback on research methodology or experiments, implement methods, assist with translation, clean and reformat dataset, support qualitative and thematic data analysis, interpret results are not applicable to this work. Additionally, we have reviewed all AI-assisted work, such as papers scraped by AI tools for literature studies. We take responsibility for the final content of this work, including text, claims or artefacts produced with the aid of generative AI.

ETHICS STATEMENT

This work does not involve human subjects, personal data, or human participant studies. We use publicly available benchmark datasets, including CIFAR-100, Tiny-ImageNet, ImageNet-100, ImageNet-R, N-Caltech101, and DVS-Gesture, for evaluating continual learning methods. The study does not collect or process personally identifiable information and does not introduce methods intended to cause physical, social, or security-related harm. We follow standard research practices for reporting experimental results and acknowledge the publicly available datasets and prior methods used for comparison. No conflicts of interest or external sponsorship influenced the design, analysis, or reporting of this work.

REPRODUCIBILITY STATEMENT

We provide detailed information to support the reproducibility of the proposed sLoTh framework and its experimental evaluation. The main paper describes the model architecture, plasticity mechanisms, training objective, classifier and prototype-based inference, and online novelty-detection procedure. The appendix provides the complete problem formulation, extended method details, training procedure, dataset descriptions, hyperparameter settings, and additional experimental results. All experiments are repeated three times using fixed random seeds. We also provide the training algorithm and implementation details needed to reproduce the continual learning experiments. We will make the code public upon acceptance.

REFERENCES

Rahaf Aljundi, Eugene Belilovsky, Tinne Tuytelaars, Laurent Charlin, Massimo Caccia, Min Lin, and Lucas Page-Caccia. Online continual learning with maximal interfered retrieval. *Advances in neural information processing systems*, 32, 2019a.

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, 2019b.

Pietro Buzzega, Matteo Boschini, Angelo Porrello, Davide Abati, and Simone Calderara. Dark experience for general continual learning: a strong, simple baseline. *Advances in neural information processing systems*, 33:15920–15930, 2020.

Arslan Chaudhry, Puneet K Dokania, Thalaiyasingam Ajanthan, and Philip HS Torr. Riemannian walk for incremental learning: Understanding forgetting and intransigence. In *Proceedings of the European conference on computer vision (ECCV)*, pages 532–547, 2018a.

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with a-gem. *arXiv preprint arXiv:1812.00420*, 2018b.

Mike Davies, Narayan Srinivasa, Tsung-Han Lin, Gautham Chinya, Yongqiang Cao, Sri Harsha Choday, Georgios Dimou, Prasad Joshi, Nabil Imam, Shweta Jain, et al. Loihi: A neuromorphic manycore processor with on-chip learning. *Ieee micro*, 38(1):82–99, 2018.

Ignacio Meza De La Jara, Cristian Rodriguez-Opazo, Damien Teney, Damith Ranasinghe, and Ehsan Abbasnejad. Mysteries of the deep: Role of intermediate representations in out of distribution detection. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025.

Jianchuan Ding, Bo Dong, Felix Heide, Yufei Ding, Yunduo Zhou, Baocai Yin, and Xin Yang. Biologically inspired dynamic thresholds for spiking neural networks. *Advances in neural information processing systems*, 35:6090–6103, 2022.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *arXiv preprint arXiv:2010.11929*, 2020.

Yuetong Fang, Deming Zhou, Ziqing Wang, Hongwei Ren, Lusong Li, Renjing Xu, et al. Spiking neural networks need high-frequency information. *Advances in Neural Information Processing Systems*, 38:103404–103425, 2026.

Yiduo Guo, Bing Liu, and Dongyan Zhao. Online continual learning through mutual information maximization. In *International conference on machine learning*, pages 8109–8126. PMLR, 2022.

Ilyass Hammouamri, Timothée Masquelier, and Dennis George Wilson. Mitigating catastrophic forgetting in spiking neural networks through threshold modulation. *Transactions on Machine Learning Research*, 2022. ISSN 2835-8856. URL https://openreview.net/forum?id=15SoThZmtU.

Bing Han, Feifei Zhao, Yi Zeng, Wenxuan Pan, and Guobin Shen. Enhancing efficient continual learning with dynamic structure development of spiking neural networks. In Edith Elkind, editor, *Proceedings of the Thirty-Second International Joint Conference on Artificial Intelligence, IJCAI-23*, pages 2993–3001. International Joint Conferences on Artificial Intelligence Organization, 8 2023. doi: 10.24963/ijcai.2023/334. URL https://doi.org/10.24963/ijcai.2023/334. Main Track.

Jiangpeng He, Zhihao Duan, and Fengqing Zhu. Cl-lora: Continual low-rank adaptation for rehearsal-free class-incremental learning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 30534–30544, 2025.

Saihui Hou, Xinyu Pan, Chen Change Loy, Zilei Wang, and Dahua Lin. Learning a unified classifier incrementally via rebalancing. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 831–839, 2019.

Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. Parameter-efficient transfer learning for nlp. In *International conference on machine learning*, pages 2790–2799. PMLR, 2019.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. *Iclr*, 1(2):3, 2022.

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

Jiashuo Li, Shaokun Wang, Bo Qian, Yuhang He, Xing Wei, Qiang Wang, and Yihong Gong. Dynamic integration of task-specific adapters for class incremental learning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 30545–30555, 2025.

Yan-Shuo Liang and Wu-Jun Li. Loss decoupling for task-agnostic continual learning. *Advances in Neural Information Processing Systems*, 36:11151–11167, 2023.

Yan-Shuo Liang and Wu-Jun Li. Inflora: Interference-free low-rank adaptation for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23638–23647, 2024.

Yan-Shuo Liang, Jia-Rui Chen, and Wu-Jun Li. Gated integration of low-rank adaptation for continual learning of large language models. *arXiv preprint arXiv:2505.15424*, 2025.

Huiwei Lin, Baoquan Zhang, Shanshan Feng, Xutao Li, and Yunming Ye. Pcr: Proxy-based contrastive replay for online class-incremental continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 24246–24255, 2023.

Sihao Liu, Yibo Yang, Xiaojie Li, David A Clifton, and Bernard Ghanem. Enhancing online continual learning with plug-and-play state space model and class-conditional mixture of discretization. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 20502–20511, 2025.

Xialei Liu, Marc Masana, Luis Herranz, Joost Van de Weijer, Antonio M Lopez, and Andrew D Bagdanov. Rotate your networks: Better weight consolidation and less catastrophic forgetting. *arXiv preprint arXiv:1802.02950*, 2018.

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

De Ma, Xiaofei Jin, Shichun Sun, Yitao Li, Xundong Wu, Youneng Hu, Fangchao Yang, Huajin Tang, Xiaolei Zhu, Peng Lin, and Gang Pan. Darwin3: A large-scale neuromorphic chip with a novel isa and on-chip learning, 2023. URL https://arxiv.org/abs/2312.17582.

Saber Moradi, Ning Qiao, Fabio Stefanini, and Giacomo Indiveri. A scalable multicore architecture with heterogeneous memory structures for dynamic neuromorphic asynchronous processors (dynaps). *IEEE transactions on biomedical circuits and systems*, 12(1):106–122, 2017.

Vaishnavi Nagabhushana, Kartikay Agrawal, and Ayon Borthakur. Catformer: When continual learning meets spiking transformers with dynamic thresholds. *arXiv preprint arXiv:2603.15184*, 2026.

Wenyao Ni, Jiangrong Shen, Qi Xu, and Huajin Tang. Alade-snn: Adaptive logit alignment in dynamically expandable spiking neural networks for class incremental learning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pages 19712–19720, 2025.

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*, pages 2001–2010, 2017.

Ole Richter, Chenxi Wu, Adrian M Whatley, German Köstinger, Carsten Nielsen, Ning Qiao, and Giacomo Indiveri. Dynap-se2: a scalable multi-core dynamic neuromorphic asynchronous spiking neural network processor. *Neuromorphic Computing and Engineering*, 4(1):014003, jan 2024. doi: 10.1088/2634-4386/ad1cd7. URL https://doi.org/10.1088/2634-4386/ad1cd7.

David Rolnick, Arun Ahuja, Jonathan Schwarz, Timothy Lillicrap, and Gregory Wayne. Experience replay for continual learning. *Advances in neural information processing systems*, 32, 2019.

Jiangrong Shen, Wenyao Ni, Qi Xu, and Huajin Tang. Efficient spiking neural networks with sparse selective activation for continual learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 38(1):611–619, Mar. 2024. doi: 10.1609/aaai.v38i1.27817. URL https://ojs.aaai.org/index.php/AAAI/article/view/27817.

Juncheng Shen, De Ma, Zonghua Gu, Ming Zhang, Xiaolei Zhu, Xiaoqiang Xu, Qi Xu, Yangjing Shen, and Gang Pan. Darwin: a neuromorphic hardware co-processor based on spiking neural networks. *Science China Information Sciences*, 59(2):1–5, 2016.

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.

Albin Soutif-Cormerais, Antonio Carta, and Joost Van de Weijer. Improving online continual learning performance and stability with temporal ensembles. In *Conference on Lifelong Learning Agents*, pages 828–845. PMLR, 2023.

Gido M Van de Ven and Andreas S Tolias. Three scenarios for continual learning. *arXiv preprint arXiv:1904.07734*, 2019.

Maorong Wang, Nicolas Michel, Ling Xiao, and Toshihiko Yamasaki. Improving plasticity in online continual learning via collaborative learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23460–23469, 2024.

Zifeng Wang, Zizhao Zhang, Sayna Ebrahimi, Ruoxi Sun, Han Zhang, Chen-Yu Lee, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, et al. Dualprompt: Complementary prompting for rehearsal-free continual learning. In *European conference on computer vision*, pages 631–648. Springer, 2022a.

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*, pages 139–149, 2022b.

Xiwen Wei, Guihong Li, and Radu Marculescu. Online-lora: Task-free online continual learning via low rank adaptation. In *2025 IEEE/CVF Winter Conference on Applications of Computer Vision (WACV)*, pages 6634–6645. IEEE, 2025.

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.

Yichen Wu, Hongming Piao, Long-Kai Huang, Renzhen Wang, Wanhua Li, Hanspeter Pfister, Deyu Meng, Kede Ma, and Ying Wei. Sd-lora: Scalable decoupled low-rank adaptation for class incremental learning. *arXiv preprint arXiv:2501.13198*, 2025.

Hongwei Yan, Liyuan Wang, Kaisheng Ma, and Yi Zhong. Orchestrate latent expertise: Advancing online continual learning with multi-level supervision and reverse self-distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23670–23680, 2024.

Jingkang Yang, Kaiyang Zhou, Yixuan Li, and Ziwei Liu. Generalized out-of-distribution detection: A survey. *International Journal of Computer Vision*, 132(12):5635–5662, 2024.

Man Yao, Xuerui Qiu, Tianxiang Hu, Jiakui Hu, Yuhong Chou, Keyu Tian, Jianxing Liao, Luziwei Leng, Bo Xu, and Guoqi Li. Scaling spike-driven transformer with efficient spike firing approximation training. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(4): 2973–2990, 2025.

Fei Ye and Adrian G Bors. Online task-free continual generative and discriminative learning via dynamic cluster memory. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 26202–26212, 2024.

Friedemann Zenke, Ben Poole, and Surya Ganguli. Continual learning through synaptic intelligence. In *International conference on machine learning*, pages 3987–3995. Pmlr, 2017.

Qingru Zhang, Minshuo Chen, Alexander Bukharin, Pengcheng He, Yu Cheng, Weizhu Chen, and Tuo Zhao. Adaptive budget allocation for parameter-efficient fine-tuning. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=lq62uWRJjiY.

Chenlin Zhou, Han Zhang, Zhaokun Zhou, Liutao Yu, Liwei Huang, Xiaopeng Fan, Li Yuan, Zhengyu Ma, Huihui Zhou, and Yonghong Tian. QKFormer: Hierarchical spiking transformer using q-k attention. In *The Thirty-eighth Annual Conference on Neural Information Processing Systems*, 2024a. URL https://openreview.net/forum?id=AVd7DpiooC.

Chenlin Zhou, Liutao Yu, Zhaokun Zhou, Han Zhang, Jiaqi Wang, Huihui Zhou, Zhengyu Ma, and Yonghong Tian. Spikingformer: A key foundation model for spiking neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 2236–2244, 2026.

Da-Wei Zhou, Hai-Long Sun, Jingyi Ning, Han-Jia Ye, and De-Chuan Zhan. Continual learning with pre-trained models: A survey. In Kate Larson, editor, *Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24*, pages 8363–8371. International Joint Conferences on Artificial Intelligence Organization, 8 2024b. doi: 10.24963/ijcai.2024/924. URL https://doi.org/10.24963/ijcai.2024/924. Survey Track.

Zhaokun Zhou, Yuesheng Zhu, Chao He, Yaowei Wang, Shuicheng YAN, Yonghong Tian, and Li Yuan. Spikformer: When spiking neural network meets transformer. In *The Eleventh International Conference on Learning Representations*, 2023. URL https://openreview.net/forum?id=frE4fUwz_h.

A Extended Method Details

This section provides additional methodology details for the **sLoTh** framework.

A.1 Problem Formulation Details

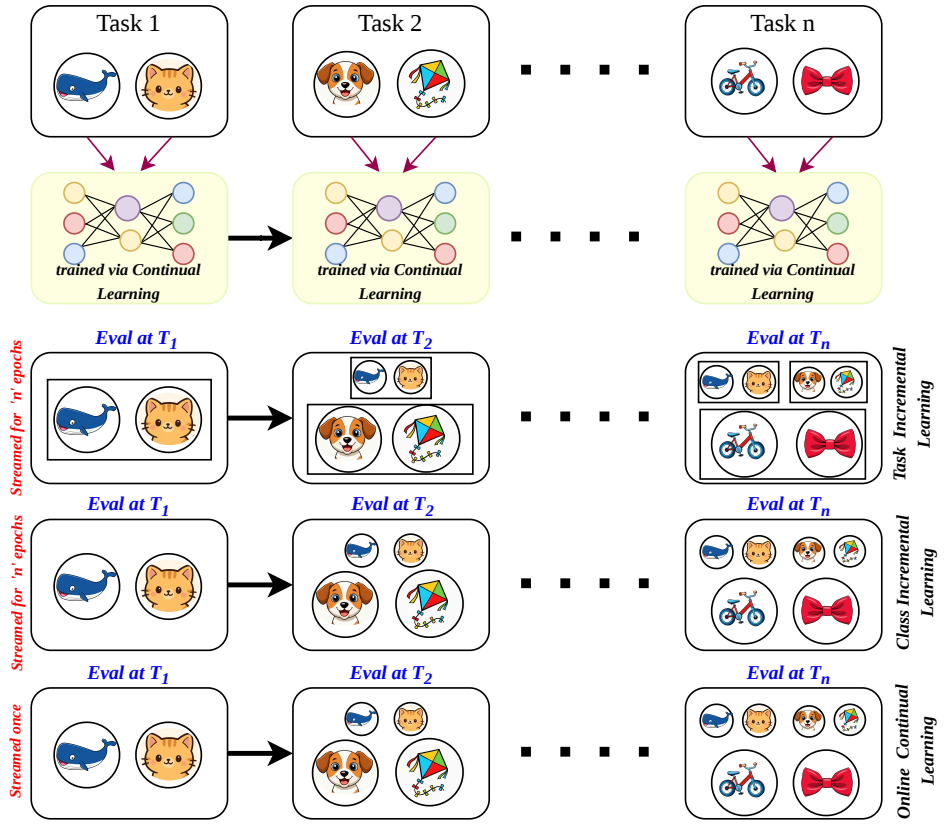

Figure 2: Comparison of Continual Learning Paradigms. In Task-Incremental Learning (**TIL**), **task boundaries are known**, allowing for task-specific components during evaluation. Class-Incremental Learning (**CIL**) increases difficulty by requiring the model to distinguish between all seen classes **without task IDs** at inference. Online Continual Learning (**OCL**) is a **challenging scenario**, as it combines the expanding output space of CIL with the constraint that data is streamed only once.

We consider a sequence of tasks $\{\mathcal{D}_t\}_{t=1}^{T}$ where each task $\mathcal{D}_t = \{(x_i, y_i)\}_{i=1}^{N_t}$ introduces a disjoint label set $\mathcal{C}_t$ such that $\mathcal{C}_i \cap \mathcal{C}_j = \varnothing$ for $i \neq j$. The cumulative label space after task t is

$$\mathcal{C}_{\leq t} = \bigcup_{k=1}^{t} \mathcal{C}_k. \tag{5}$$

Given an input x, the model predicts

$$\hat{y} \in \mathcal{C}_{\leq t}. \tag{6}$$

As part of our study, we evaluate **Full sLoTh**, which comprises two components, i.e seLoRA and Threshold modulation, under three protocols:
OCL: Training data arrive as a stream $\{(x_{t,b}, y_{t,b})\}$ with each sample observed once and no task identity t is provided. The model must autonomously detect distribution shifts and preserve knowledge without explicit task boundaries.
CIL: Given tasks $\{D_t\}_{t\in\mathcal{T}}$ with $\mathcal{C}_i \cap \mathcal{C}_j = \varnothing$ for $i \neq j$, the task identity is unknown at test time; predictions satisfy $\hat{y} \in \mathcal{C}_{\leq t} = \bigcup_{i=1}^{t} \mathcal{C}_i$.
TIL: Given tasks $\{D_t\}_{t\in\mathcal{T}}$ with $p(\mathcal{X}_i) \neq p(\mathcal{X}_j)$ and $\mathcal{C}_i \cap \mathcal{C}_j = \varnothing$ for $i \neq j$, the task identity t is known at test time; predictions satisfy $\hat{y} \in \mathcal{C}_t$.

In the OCL setting, explicit task boundaries are not provided to the model. Instead, task transitions are implicitly inferred through the proposed novelty detection mechanism (Section 3.5). When a significant distribution shift is detected, the model treats this event as a pseudo-task boundary, triggering parameter snapshotting and classifier expansion. Therefore, while no ground-truth task identity is available, the framework internally constructs adaptive boundaries based on data-driven signals.

A.2 Plasticity Modules

The plasticity mechanisms are selected according to the continual learning regime. For CIL and TIL, where task boundaries are known, and each task is observed for multiple epochs, we jointly optimize threshold modulation and seLoRA, i.e., $\psi = \delta, \phi$. For OCL, where samples are observed only once and task boundaries are unavailable, we evaluate Full sLoTh with joint adaptation as well as the corresponding single-mechanism variants, $\psi = \delta$ and $\psi = \phi$. The latter variants isolate threshold modulation and seLoRA, respectively, and allow us to study their behavior under single-pass adaptation. The backbone network is a pretrained, spiking transformer $f_\theta(\cdot)$ with frozen parameters θ. Plasticity is restricted to a small set of adaptation parameters ψ.

$$z = f_{\theta,\psi}(x) \tag{7}$$

, where $z \in \mathbb{R}^d$ is the feature representation. Depending on the continual learning setting, we use:

$$\psi = \begin{cases} \{\phi, \delta\},\ \{\delta\},\ \text{or } \{\phi\} & \text{for OCL} \\ \{\phi, \delta\} & \text{for CIL or TIL} \end{cases} \tag{8}$$

where δ represents threshold modulation parameters and ϕ denotes low-rank attention parameters.

A.2.1 Channel-wise Excitability Modulation

In the Leaky-Integrate and Fire neuron, when the membrane potential exceeds a threshold v_{th} the neuron produces spikes. To make the thresholds learnable instead of utilising the intrinsic dynamic and backpropagating on the thresholds itself we introduce channel-wise offsets δ_c:

$$v_{\text{th}}(c) = v_{\text{th}}^{base} + \delta_c \tag{9}$$

where c indexes feature channels. This mechanism helps modulate neuronal excitability while keeping the underlying synaptic weights frozen. Hence, the feature geometry produced by the pretrained backbone remains largely preserved.

A.2.2 SCALABLE-EFFICIENT LORA (SELORA)

We additionally apply low-rank attention modulation to provide a complementary source of plasticity to threshold modulation. For each attention projection matrix $W \in \mathbb{R}^{d\times d}$, a low-rank residual update is introduced:

$$W' = W + BA, \tag{10}$$

where $B \in \mathbb{R}^{d\times r}$ and $A \in \mathbb{R}^{r\times d}$, $\quad r \ll d$ and $\phi = \{A_i, B_i\}$ are the learnable parameters. Unlike task-specific LoRA modules proposed in prior work Wu et al. (2025); Liang and Li (2024); He et al. (2025), seLoRA uses a single shared low-rank module across the task sequence. This prevents parameter growth as the number of tasks increases.

A.3 STABILIZED TRAINING OBJECTIVE

The overall training objective is

$$\mathcal{L} = \mathcal{L}_{CE} + \mathcal{L}_{stab} \tag{11}$$

where

$$\mathcal{L}_{stab} = \lambda_{KL}\mathcal{L}_{KL} + \lambda_{local}\mathcal{L}_{local} + \lambda_{global}\mathcal{L}_{global}. \tag{12}$$

Cross-Entropy Loss The classifier is trained using cross-entropy over the current task classes:

$$\mathcal{L}_{CE} = CE(\ell_{\mathcal{C}_t}(x), y). \tag{13}$$

Logit Distillation To preserve knowledge of previous classes, we distil predictions from a frozen teacher model:

$$\mathcal{L}_{KL} = \tau^2 KL\left(softmax(\ell^s/\tau) \parallel softmax(\ell^*/\tau)\right) \tag{14}$$

where τ is the temperature parameter.

Local Parameter Anchor To prevent abrupt parameter drift, we anchor adaptation parameters to their previous snapshot:

$$\mathcal{L}_{local} = ||\psi - \psi^*||_2^2. \tag{15}$$

Global Parameter Anchor To prevent long-term drift across many tasks, we regularise parameters toward their initial state:

$$\mathcal{L}_{global} = \frac{1}{|\psi|}\sum_{p\in\psi}(p - p^{(0)})^2. \tag{16}$$

A.4 PROTOTYPE-BASED INFERENCE

During training, a cosine classifier is used:

$$\ell_c(x) = s \cdot \frac{z}{||z||_2} \cdot \frac{w_c}{||w_c||_2} \tag{17}$$

After training, predictions are made using Nearest Class Mean classification. Class prototypes are computed as

$$\mu_c = \frac{1}{N_c}\sum_{y_i=c} f_{\theta,\psi}(x_i). \tag{18}$$

Inference is performed via cosine similarity:

$$\hat{y} = \arg\max_{c \in \mathcal{C}_{\leq t}} \cos(f_{\theta,\psi}(x), \mu_c). \tag{19}$$

A.5 Online Novelty Detection

To detect distribution shifts in OCL, we monitor the training loss. We maintain an exponential moving average

$$\bar{\mathcal{L}}_b = (1-\alpha)\bar{\mathcal{L}}_{b-1} + \alpha\mathcal{L}_b. \tag{20}$$

A novelty trigger is activated when

$$r_b = \frac{\mathcal{L}_b}{\bar{\mathcal{L}}_b} > \gamma. \tag{21}$$

When triggered, the model:

1. snapshots current adaptation parameters
2. registers newly discovered classes
3. updates the teacher model for distillation

A.6 Full Training Procedure

Algorithm 1 sLoTh Training Procedure

```
Initialize pretrained backbone f_θ and adaptation parameters ψ
Initialize EMA loss L̄_0
for each batch (x_b, y_b) do
    z ← f_{θ,ψ}(x_b)
    Compute loss L_b
    Update EMA L̄_b
    if L_b / L̄_b > γ then
        snapshot teacher parameters
    end if
    Update adaptation parameters ψ
end for
```

Algorithm 1 summarises the full training framework for OCL. Instead of triggers, we have an exclusive snapshot captured for adaptation after every task in the case of CIL and TIL

A.7 Component Analysis of Loss Function

Table 7 analyses the contribution of individual components of the proposed loss. Removing the KL regularisation term ($\lambda_{KL} = 0$) reduces accuracy from **55.12%** to **10.23%**, while removing the local and global parameter anchors reduces it to **21.64%** and **20.11%**, respectively. These results indicate that all three regularisation terms contribute to the performance of the full objective, with KL regularisation having the largest effect in the evaluated settings.

Table 7: Component analysis of Loss function on CIFAR-100 dataset in OCL setting

Ablated Components	Full Objective	$\lambda_{KL} = 0$	$\lambda_{local} = 0$	$\lambda_{global} = 0$
QKFormer + Loss Based	**55.12 ± 1.18**	10.23 ± 1.10	21.64± 2.38	20.11 ± 1.09
MaxFormer + RT	**39.60± 2.07**	9.44 ± 1.61	12.3±1.05	23.57 ± 1.12

B DATASETS

We conduct evaluations across different datasets, including CIFAR-100, Tiny-ImageNet, ImageNet-100, and ImageNet-R. For CIFAR-100, we use the standard dataset provided through the `torchvision` library. Tiny-ImageNet is obtained from the official repository.[1] ImageNet-100[2] and ImageNet-R[3] are subsets of the ImageNet-1K dataset that provide a reduced-scale benchmark and a domain-shift benchmark. To simulate continual learning scenarios, each dataset is partitioned into disjoint class subsets that form incremental tasks. For every experiment, class labels are remapped to ensure consistent label ordering across tasks. To maintain reproducibility and ensure fair evaluation, all random operations, including class permutations and data shuffling, are controlled using fixed random seeds.

C HYPERPARAMETER SETTINGS

Most hyperparameters follow the official QKFormer implementation.[4] In the online continual learning (OCL) setting, we additionally perform a small hyperparameter search over the novelty-detection threshold used to trigger teacher model snapshots. We observe that very small thresholds lead to frequent false triggers, which increase computational overhead but do not significantly affect final performance. In contrast, overly large thresholds delay the detection of distribution shifts, reducing the model's ability to effectively adapt to newly introduced classes. In practice, we select a threshold value that balances timely detection of distribution shifts with stable training dynamics.

Table 8: Hyperparameters used across datasets and continual learning settings (OCL, CIL, and TIL).

Hyperparameter	**CIFAR-100**	**Imagenet-R**	**Imagenet-100**	**Tiny-IM**
Batch size	128	64	64	128
Time steps (T)	4	4	4	4
Learning rate	1e-4	3e-4	3e-4	1e-4
Epochs	50	30	30	30
Optimizer	AdamW	AdamW	AdamW	AdamW

Each experiment is conducted three times with different seeds on the same hyperparameters, using seeds 0, 1, and 2. The following tables describe the hyperparameters used for each dataset.

D EXTENDED RESULTS

D.1 PEFT FOR SPIKING VISION TRANSFORMERS

Before introducing sLoTh, we first evaluate which PEFT mechanisms are compatible with spiking transformers. Table 9 reports the classification accuracy of different PEFT strategies applied to a QKFormer backbone on CIFAR-100. Adapter-based Houlsby et al. (2019), i.e., tuning of small feed-forward networks acting as bottleneck layers, achieves the highest accuracy (**87.21%**), indicating that additional learnable modules can effectively adapt feature representations. However, adapters introduce task-specific parameters and increase memory and inference overhead, making them less suitable for online continual learning scenarios with strict resource constraints.

Table 9: PEFT feasibility on pretrained QKFormer evaluated on the CIFAR-100.

Method	LoRA (Hu et al., 2022)	Adapters (Houlsby et al., 2019)	Spiking Adapters	Thresholds
Accuracy	80.89	**87.21**	79.82	70.56

Low-rank attention modulation using LoRA provides a more efficient alternative, achieving **80.89%** accuracy while requiring only lightweight updates to attention projections. Threshold-only adapta-

[1] `http://cs231n.stanford.edu/tiny-imagenet-200.zip`
[2] `https://www.kaggle.com/datasets/ambityga/imagenet100`
[3] `https://github.com/hendrycks/imagenet-r.git`
[4] `https://github.com/zhouchenlin2096/QKFormer.git`

tion is the most parameter-efficient strategy, modifying spiking neuron firing thresholds without introducing additional weights, and still achieves **70.56%** accuracy.

These results reveal a trade-off: adapters offer strong performance but incur architectural overhead, while threshold adaptation is extremely efficient but limited in representational flexibility. Motivated by this observation, we propose the **sLoTh framework**, which combines **low-rank attention modulation** with **threshold-based intrinsic plasticity** to achieve efficient adaptation without affecting inference efficiency.

D.2 Task-Incremental Learning and Class Incremental Learning

In addition to the online continual learning (OCL) and class-incremental learning (CIL) evaluations presented in the main paper, we report results under the Task-Incremental Learning (TIL) protocol. In this setting, task boundaries are available during both training and evaluation, making the problem comparatively easier than CIL and OCL.

Table 10: Task-Incremental Learning results for sLoTh across different datasets. Accuracy reported as mean ± std over 3 runs.

Tasks	Tiny-ImageNet	CIFAR-100	ImageNet-R	Imagenet-100
10	85.23 ± 2.94	87.38 ± 1.63	86.77 ± 1.08	89.33 ± 1.78
20	87.17 ± 1.38	86.84 ± 2.54	88.45 ± 2.79	90.92 ± 1.63
50	96.50 ± 1.25	91.20 ± 2.71	93.61 ± 1.81	94.5 ± 1.95

Table 10 reports the performance of the proposed sLoTh framework on Tiny-ImageNet, CIFAR-100, Imagenet100, and ImageNet-R across different task granularities. The results show that sLoTh maintains strong performance even as the number of tasks increases.

Table 11: CIFAR100 CIL comparison between pretrained ViT with PEFT Continual Learning methods v/s pretrained QKFormer & SpikingFormer on **Full sLoTh** for different task granularity, i.e., T represents the number of tasks. Parameters are in millions. **First** and Second best models are highlighted.

Methods	**T=10**	**T=20**	**T=50**	**# Total Param.**	**#updates**
L2P Wang et al. (2022b)	83.18±1.20	79.51±0.67	67.95±2.12	172	**0.12**
DualPrompt Wang et al. (2022a)	81.48±0.86	80.44±1.38	72.5±1.08	172	0.86
CODA-Prompt Smith et al. (2023)	86.31±0.12	81.36±0.88	73.77±0.98	172	4.6
InfLoRA Liang and Li (2024)	86.75±0.35	80.97±0.74	70.68±1.26	172	0.51
SD-LoRA Wu et al. (2025)	**88.01±0.31**	OOM	OOM	172	0.39
Full sLoTh	81.23±1.74	**84.65±1.46**	**87.22±1.59**	**64.32**	0.44
Full sLoTh on SpikingFormer	63.28±1.39	67.23± 2.03	70.48±1.21	66.34	0.56

D.3 Online Continual Learning

We evaluate simplified variants of the proposed framework on the ImageNet-100 benchmark with 50 tasks. Table 12 reports the performance of sLoTh when only seLoRA or only threshold modulation is used. Interestingly, the model achieves higher accuracy, with **seLoRA-only** reaching **70.00±3.85**

Table 12: Extended comparison on ImageNet-100 with 50 tasks. Accuracy reported as mean ± std over 3 runs.

Method	Accuracy (%)
L2P	56.25 ± 0.45
MVP	54.33 ± 3.05
Online LoRA	60.38 ± 2.59
Full sLoTh	62.14 ± 1.30
sLoTh (Threshold only)	66.91 ± 1.56
sLoTh (seLoRA only)	**70.00 ± 3.85**

and **threshold-only** reaching **66.91±1.56**. This behaviour is consistent with our observations in

the online continual learning (OCL) setting given in the main paper's Table 4, where highly plastic mechanisms can adapt quickly in streaming scenarios. We also evaluate the **seLoRA-only** variant of our method on a pretrained ViT-B/16 backbone (86M parameters) for 100 tasks on Tiny-ImageNet. The corresponding accuracy is reported in Table 12.

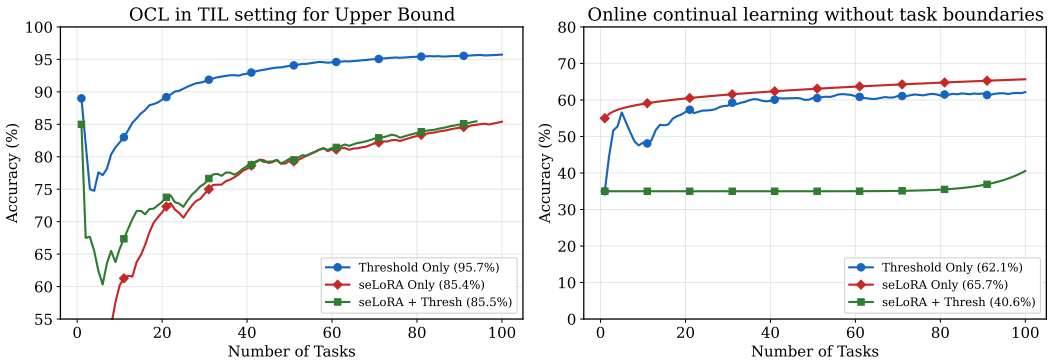

Figure 3: Performance gap between boundary-aware OCL v/s boundary-free OCL setting on Tiny-Imagenet. Here, sLoTh narrows the gap between these two settings compared to other methods without relying on replay.

E Hardware and Software Environment

All experiments were conducted on enterprise-grade workstation GPUs with hardware acceleration via CUDA 12.8. The training infrastructure consists of a workstation running Ubuntu 22.04.5 LTS with Python 3.13.11. Our implementation is based on PyTorch 2.10.0 and torchvision 0.25.0. To improve training stability and computational efficiency, we employed mixed-precision training via automatic mixed-precision (AMP).