CL-MAMBA: CONTINUAL LEARNING MAMBA SUB-NETWORKS FOR ROBUST CROSS-DOMAIN HIGH EFFICIENT HYPERSPECTRAL ANOMALY DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Bio-inspired continual learning draws inspiration from biological neural systems that continuously acquire new knowledge while retaining previously learned information. In particular, the Drosophila mushroom body employs parallel and compartmentalized neural pathways to encode complementary information and reduce interference across experiences. Inspired by this mechanism, we investigate whether a collaborative multi-learner architecture can benefit hyperspectral anomaly detection (HAD) in continuously changing scenes, where models must adapt to new background distributions while avoiding catastrophic forgetting and excessive storage of historical samples. Therefore, we propose CL-SSMamba, a bio-inspired continual learning framework built upon multiple Spatial-Spectral Mamba (SSMamba) subnetworks for open-scenario hyperspectral anomaly detection (OHAD). CL-SSMamba employs parallel SSMamba learners to collaboratively reconstruct the hyperspectral background while retaining only model parameters from previous tasks, thereby reducing storage overhead and alleviating catastrophic forgetting. Each learner integrates Global Spatial Mamba (GSM) and Local Spectral Mamba (LSM) to capture long-range spatial dependencies and cross-band spectral correlations, together with orthogonal spectral correlation enhancement to suppress redundant responses. To alleviate background–anomaly imbalance, Simple Linear Iterative Clustering (SLIC) and Local Density Peak Clustering (LDPC) are introduced to enhance spatially coherent background structures and improve background–anomaly separation. In addition, reconstruction, ensemble cooperation, and continual-learning losses are jointly optimized to promote inter-learner consistency and preserve historical knowledge. Experiments on the Abu-Beach-Urban and HAD100 datasets demonstrate that CL-SSMamba effectively adapts to sequential hyperspectral scenes without replaying historical samples, providing a robust solution for OHAD in continuously changing environments.

1 INTRODUCTION

Continual learning (CL) aims to enable models to continually acquire new knowledge from sequentially arriving data while preserving previously learned information (De Lange et al., 2022). A fundamental challenge in CL is the stability–plasticity dilemma, where models need to adapt to new tasks while preventing catastrophic forgetting (CF) of previously acquired knowledge. Existing CL methods are commonly categorized into replay-based, regularization-based, and architecture-based strategies (De Lange et al., 2022). Among them, replay-based methods achieve effective knowledge retention by revisiting historical samples, but the increasing storage requirement limits their application to long-term learning scenarios with high-dimensional data. To reduce the dependence on storing raw observations, generative continual learning has emerged as an alternative paradigm, which preserves previous knowledge by modeling and reconstructing historical data distributions. Deep Generative Replay (DGR) introduced a generator–solver framework to synthesize previous-task samples during sequential learning (Shin et al., 2017), while subsequent studies further explored generative replay mechanisms inspired by biological memory systems (van de Ven et al.,

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.

2020). However, existing generative replay approaches still introduce additional computational and memory costs, especially when dealing with complex high-dimensional data.

Hyperspectral anomaly detection (HAD) provides a representative scenario for studying continual generative learning. Different from conventional recognition tasks, HAD detects rare and unknown targets by learning normal background distributions rather than relying on predefined anomaly categories. Existing autoencoder-based and GAN-based HAD methods detect anomalies through deviations from reconstructed background representations (Hu et al., 2022; Wang et al., 2022; 2023b). However, when hyperspectral scenes are sequentially observed, background distributions continuously evolve due to variations in sensors, illumination conditions, atmospheric effects, acquisition times, and geographical regions. Consequently, continual HAD requires models to continuously update background representations while retaining previously acquired knowledge from earlier domains. Recent generative continual learning methods for anomaly detection explore diffusion-based replay strategies (Li et al., 2025; Hu et al., 2025), CL-CaGAN integrates adversarial background modeling with continual learning strategies (Wang et al., 2024), and CL-BioGAN further introduces biologically inspired generative learning to improve stability and adaptability across sequential hyperspectral scenes (Wang et al., 2025). Nevertheless, these approaches may still rely on replay mechanisms, auxiliary generative models, or complex regularization, resulting in considerable computational overhead. Moreover, the high spectral dimensionality, spectral redundancy, and severe background-anomaly imbalance of HSI further increase the difficulty of efficient continual HAD.

To address these challenges, we explore efficient sequence modeling architectures for continual hyperspectral representation learning. Mamba, based on selective state-space models, provides linear-complexity sequence modeling and input-dependent state selection, enabling efficient long-range dependency modeling under limited computational resources (Gu & Dao, 2024). However, directly applying a single Mamba backbone to continual HAD may be faced with different background knowledge into shared parameters, leading to representation interference and knowledge forgetting. Inspired by the parallel and compartmentalized information-processing mechanism of the Drosophila mushroom body, we propose collaborative Mamba subnetworks to learn complementary background representations while maintaining knowledge stability. This design combines the efficient spatial-spectral modeling capability of Mamba with biologically inspired compartmentalized learning for replay-free continual HAD.

Based on this motivation, we propose CL-SSMamba, a bio-inspired continual spatial-spectral Mamba framework for open-scenario hyperspectral anomaly detection (OHAD). Instead of employing a single reconstruction model or explicitly replaying historical hyperspectral observations, CL-SSMamba constructs K parallel SSMamba learners, where each learner acts as a functional memory compartment for capturing complementary background representations. Only model parameters from previous tasks are retained, avoiding the storage of historical HSIs. Within each learner, Global Spatial Mamba (GSM) captures long-range spatial dependencies, while Local Spectral Mamba (LSM) models local and remote spectral correlations across bands. A Spectral Correlation Enhancement (SCE) module further reduces spectral redundancy through orthogonal spectral representation learning. Considering the severe imbalance between background and anomalous pixels, Simple Linear Iterative Clustering (SLIC) and Local Density Peak Clustering (LDPC) are introduced to enhance spatially coherent background structures and improve background–anomaly separability. Finally, reconstruction loss, ensemble cooperation loss, and continual-learning regularization are jointly optimized to encourage learner collaboration and preserve previously acquired knowledge during sequential adaptation.

The main innovations and contributions are summarized as follows:

- Inspired by the memory-efficient biological structure, a novel CL-SSMamba is proposed by integrating continual spatial-spectral Mamba subnetworks and a collaborative multi-learner paradigm, which can process sequential hyperspectral data streams while mitigating catastrophic forgetting without storing historical HSIs.
- A robust hybrid sampling strategy integrating SLIC and LDPC is introduced to construct more reliable background distributions and suppress background overfitting in imbalanced anomaly detection scenarios.
- A parallel-sequential spatial-spectral Mamba architecture is designed by integrating Global Spatial Mamba (GSM), Local Spectral Mamba (LSM), and Spectral Correlation Enhance-

ment (SCE) to capture long-range spatial dependencies and reduce spectral redundancy with linear-complexity state-space modeling.

- A synthesized loss composed of reconstruction loss L_{recon}, ensemble cooperation loss L_{EC}, and continual learning loss L_{CL} is designed to jointly refine background reconstruction, learner cooperation, and historical-knowledge retention.

2 Related Work

2.1 Traditional Hyperspectral Anomaly Detection

Traditional HAD methods identify anomalous pixels by modeling statistical, geometric, or structural characteristics of the background without requiring prior target signatures. Classical statistical detectors, such as RX and kernel RX, estimate background distributions and identify spectrally deviating observations (Chang & Chiang, 2002; Kwon & Nasrabadi, 2005). Recent model-driven methods improve background characterization through low-rank and sparse modeling or discriminative spectral selection, such as HADGSM (Ren et al., 2024) and rapid discriminative band selection (Yan et al., 2024). Although these approaches generally provide good interpretability, their reliance on fixed statistical assumptions or manually designed priors limits their adaptability to continuously changing scenes.

2.2 Deep Learning-Based Hyperspectral Anomaly Detection

Deep learning has advanced HAD by learning nonlinear spatial-spectral representations directly from hyperspectral observations. Reconstruction-based methods, such as Auto-AD and SSCMAE, model dominant background structures and detect anomalies from reconstruction discrepancies (Wang et al., 2022; Guo et al., 2024). Self-supervised and direct-detection paradigms further improve feature separability, as demonstrated by MSNet (Liu et al., 2024) and the one-step spectral-deviation detector (Li et al., 2024). Recently, state-space models have shown promise for efficient long-range representation learning in hyperspectral imagery (Pan et al., 2025) and anomaly detection (He et al., 2024; 2025). Nevertheless, these methods are predominantly developed under static training settings and do not address knowledge retention across sequential hyperspectral domains.

2.3 Continual Learning-Based Hyperspectral Anomaly Detection

Continual learning methods are categorized into regularization-based, replay-based, and architecture-based approaches (De Lange et al., 2022). Regularization methods such as EWC constrain parameter updates to preserve previously acquired knowledge (Kirkpatrick et al., 2017), whereas replay-based approaches such as iCaRL and A-GEM revisit representative historical samples or episodic memories (Rebuffi et al., 2017; Chaudhry et al., 2019). Architecture-based methods instead reduce task interference through dedicated or dynamically generated parameter spaces; representative examples include continual learning with hypernetworks (von Oswald et al., 2020) and forget-free winning subnetworks (Kang et al., 2025). Recent replay-free studies further explore fixed-capacity gradient optimization (Liu & Liu, 2022) and decomposed prompting (Smith et al., 2023). In addition, BatchEnsemble demonstrates that parameter-sharing ensembles can support efficient lifelong learning with substantially reduced memory and computational costs (Wen et al., 2020).

Applying these strategies directly to OHAD remains challenging because sequential hyperspectral scenes exhibit substantial domain shifts while anomaly labels are generally unavailable. CL-CaGAN combines adversarial background modeling with continual learning for cross-domain HAD (Wang et al., 2024), while CL-BioGAN introduces biologically inspired mechanisms to balance stability and plasticity across hyperspectral scenes (Wang et al., 2025). Nevertheless, existing continual HAD methods may still depend on replayed observations or auxiliary generative mechanisms. This motivates a replay-free architecture that can maintain historical background knowledge while efficiently adapting to evolving hyperspectral domains.

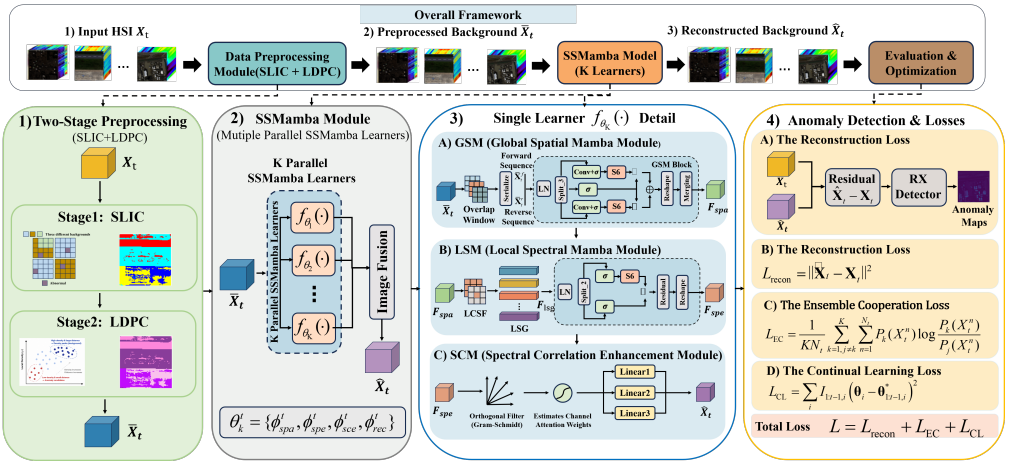


Figure 1: The specific structural diagram of the proposed CL-SSMamba.

3 METHODOLOGY

In this section, we introduce the designed CL-SSMamba for continual OHAD. The overall schematic is illustrated in Fig. 1. CL-SSMamba integrates two-stage data preprocessing, a bio-inspired collaborative multi-learner paradigm with an SSMamba-based feature extraction backbone, and continual loss optimization to mitigate catastrophic forgetting in sequential OHAD tasks. Multiple parallel learners collaboratively extract complementary spatial and spectral representations, while spectral information within each learner is progressively refined through local spectral grouping, long-range spectral dependency modeling, and spectral correlation enhancement. During sequential learning, only the learner parameters retained from the preceding task are preserved and updated, without storing historical hyperspectral samples for replay.

3.1 DATA PREPROCESSING MODULE (SLIC + LDPC)

The data preprocessing module contains two consecutive stages rather than two independent clustering operations. In the first stage, Simple Linear Iterative Clustering (SLIC) partitions the HSI into spatially compact and spectrally homogeneous superpixels. The purpose of this stage is not to directly identify anomalous pixels. Instead, it increases the local proportion of a spatially coherent but globally sparse background category within an individual superpixel. This local enhancement makes the sparse background more distinguishable from isolated anomalies.

In the second stage, Local Density Peak Clustering (LDPC) is applied to the superpixel-level representations generated by SLIC. LDPC evaluates the local density and relative distance of each superpixel. Superpixels associated with dense and repeatedly occurring structures are assigned to background clusters, whereas isolated or low-density superpixels are treated as anomaly candidates. Therefore, SLIC establishes reliable local regions, and LDPC subsequently performs background-anomaly separation at the region level.

At SLIC iteration v, let $\boldsymbol{\mu}_m^{(v)}$ and $\mathbf{c}_m^{(v)}$ denote the spectral centroid and spatial center of superpixel m. For each pixel, $\mathbf{x}_i$ is its spectral vector and $\mathbf{p}_i$ its two-dimensional spatial coordinate. The composite distance between pixel i and superpixel m is

$$d_{i,m}^{(v)} = \left\|\mathbf{x}_i - \boldsymbol{\mu}_m^{(v)}\right\|^2 + \lambda_{\text{sp}} \left\|\mathbf{p}_i - \mathbf{c}_m^{(v)}\right\|^2, \tag{1}$$

where λ_{sp} is the spatial-distance weighting factor.

After SLIC convergence, each superpixel is represented by its mean spectral vector, and LDPC further performs region-level background-anomaly separation. The optimized tensor dataset is represented as $\bar{\mathbf{X}}_t$ for overlapping-patch generation and subsequent background reconstruction. The initialization, center-update process, and complete preprocessing procedure are provided in Appendix A.3.

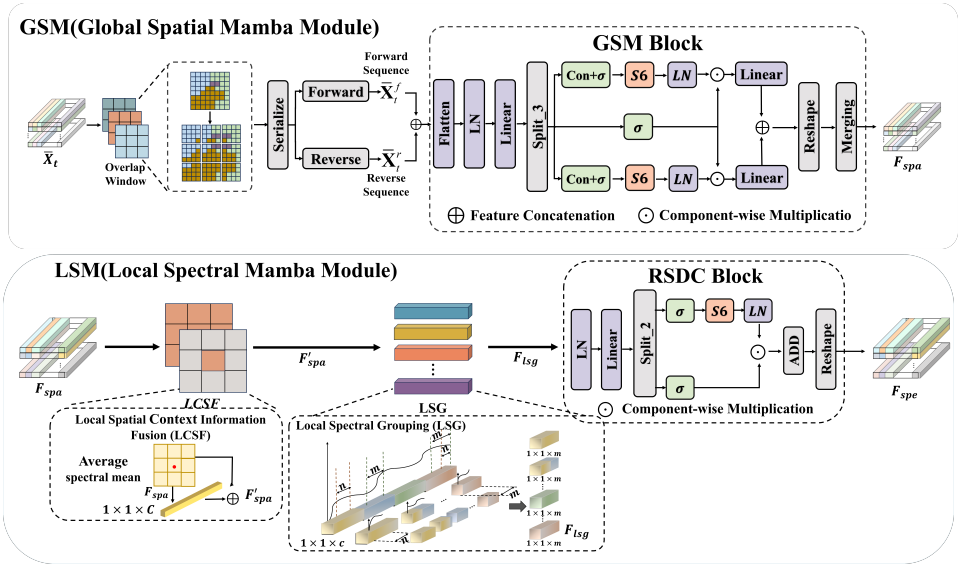


Figure 2: The detailed structures of the proposed Global Spatial Mamba (GSM) and Local Spectral Mamba (LSM) feature extraction modules.

3.2 The Structure of SSMamba

The SSMamba mainly consists of K parallel learners, and each learner is sequentially comprised of Global Spatial Mamba (GSM), Local Spectral Mamba (LSM), Spectral Correlation Enhancement (SCE), and a reconstruction head. The detailed structures of GSM and LSM are illustrated in Fig. 2.

Multiple Parallel SSMamba Learners. Motivated by neuro-inspired continual learning studies that highlight parallel modularity as an effective mechanism for reducing knowledge interference (Wang et al., 2023a), together with the coordinated and compartmentalized neuromodulation observed in the Drosophila mushroom body (Cohn et al., 2015), CL-SSMamba adopts a parallel multi-learner architecture rather than the conventional single-learner paradigm. Specifically, the k-th learner is instantiated as an independently parameterized SSMamba subnetwork $f_{\theta_k}(\cdot)$, allowing different learners to capture complementary background representations during sequential OHAD tasks. The outputs of all learners are subsequently aggregated to produce the final reconstructed background image:

$$\hat{\mathbf{X}}_t = \sum_{k=1}^{K} s \cdot f_{\theta_k}(\bar{\mathbf{X}}_t), \tag{2}$$

where $\bar{\mathbf{X}}_t$ denotes the preprocessed input of task t. The complete set of trainable parameters in each learner is

$$\boldsymbol{\theta}_k^t = \{\boldsymbol{\phi}_{spa}^t, \boldsymbol{\phi}_{spe}^t, \boldsymbol{\phi}_{sce}^t, \boldsymbol{\phi}_{rec}^t\}, \tag{3}$$

where $\boldsymbol{\phi}_{spa}^t, \boldsymbol{\phi}_{spe}^t, \boldsymbol{\phi}_{sce}^t$ and $\boldsymbol{\phi}_{rec}^t$ respectively denote the sequential Mamba learning parameters for Global Spatial Mamba Module (GSM) , Local Spectral Mamba Module (LSM), Spectral Correlation Enhancement Module (SCE), and the reconstruction head for the current t-th learning HAD task.

Global Spatial Mamba Module (GSM). In view of Mamba can model long-range dependencies in linear complexity, the proposed GSM module is designed to capture global spatial structural information within the image. To mitigate the disruption of local neighborhood pixel correlations and the inadequacy of spatial feature representation inherent in traditional patch generation, we employ an overlapping-window strategy with stride set to half the patch size. The processed tensor $\bar{\mathbf{X}}_t$ is serialized into a forward sequence $\bar{\mathbf{X}}_t^f$ and a reverse sequence $\bar{\mathbf{X}}_t^r$, both of which are fed into the Mamba block for bidirectional modeling.

After layer normalization (LN) and linear transformation (Linear), the bidirectional sequence is split into forward, reverse, and gating branches:

$$\tilde{\mathbf{X}}_t^f, \tilde{\mathbf{X}}_t^r, \tilde{\mathbf{X}}_t^z = \mathrm{Split}_3\Big(\mathrm{Linear}\Big(\mathrm{LN}(\mathrm{Flatten}(\bar{\mathbf{X}}_t^f \oplus \bar{\mathbf{X}}_t^r))\Big)\Big). \tag{4}$$

where $\oplus$ denotes feature concatenation, and $\mathrm{Split}_3(\cdot)$ denotes an equal partition of the projected feature along the channel dimension into three sub-features.

The forward and reverse branches are subsequently processed by convolution (Conv), SiLU activation (σ), and the S6 operator:

$$\tilde{\mathbf{X}}_t^{d\prime} = \mathrm{S6}\Big(\sigma(\mathrm{Conv}(\tilde{\mathbf{X}}_t^d))\Big), \quad d \in \{f, r\}. \tag{5}$$

The processed bidirectional sequence are gated, projected, and merged to obtain the global spatial feature $\mathbf{F}_{spa}$. The complete GSM formulation is given in Appendix A.3.

Local Spectral Mamba Module (LSM). Local spectral features are instrumental in capturing subtle variations and provide contextual information for differentiating spectral characteristics across diverse samples. Distinct from GSM, the LSM module is specifically engineered for the joint extraction of spectral and local spatial features. The process involves three key steps. First, Local Spatial Context Fusion (LCSF) calculates the local mean $\mathbf{m}_{i,j}$ using a $w \times w$ sliding window and fuses it with the original pixel feature:

$$\mathbf{m}_{i,j} = \frac{1}{w^2} \sum_{i'=i-w/2}^{i+w/2} \sum_{j'=j-w/2}^{j+w/2} \mathbf{F}_{i',j'} \tag{6}$$

fused with the original pixel features to obtain

$$\mathbf{f}'_{i,j} = \mathbf{f}_{i,j} + \mathbf{m}_{i,j} \tag{7}$$

and subsequently generate $\mathbf{F}'_{\mathrm{spa}}$.

Second, Local Spectral Grouping (LSG) groups the spatial feature along the spectral dimension to reduce redundancy:

$$g = \lfloor \frac{C-(m-n)}{n} \rfloor + \lceil \frac{(c-(m-n)) \bmod n + m - n}{n} \rceil \tag{8}$$

where C denotes the number of channels, m and n respectively denote the spectral-group width and group stride, and outputs $\mathbf{F}_{\mathrm{lsg}}$.

Third, Remote Spectral Dynamic Correlation (RSDC) models long-range dependencies among spectral groups.

Specifically, $\mathbf{F}_{\mathrm{lsg}}$ is first normalized and linearly projected, and then divided into the main spectral branch $\mathbf{F}_s$ and the gating branch $\mathbf{F}_z$ as:

$$\mathbf{F}_s, \mathbf{F}_z = \mathrm{Split}_2\left(\mathrm{linear}\left(\mathrm{LN}(\mathbf{F}_{\mathrm{lsg}})\right)\right) \tag{9}$$

where $\mathrm{Split}_2(\cdot)$ partitions the projected feature into two sub-features along the channel dimension. The main spectral feature is subsequently processed by convolution, SiLU activation function (σ), and the S6 operator.

$$\mathbf{F}_{S6} = S6\left(\sigma(\mathbf{F}_s)\right) \tag{10}$$

The state-space representation is then modulated by the gating feature and combined with the residual input to obtain the spectral representation:

$$\mathbf{F}_{\mathrm{spe}} = \mathrm{Reshape}(\mathrm{linear}(\mathrm{LN}(\mathbf{F}_{S6}) \odot \mathrm{Silu}(\mathbf{F}_z)) + \mathbf{F}_{\mathrm{lsg}}) \tag{11}$$

Here, $\mathbf{F}_{spe}$ represents the output spectral feature of RSDC.

Spectral Correlation Enhancement Module (SCE). Adjacent hyperspectral bands are highly correlated and frequently contain redundant information. To reduce this redundancy, the filter set $\{\mathbf{FIL}_c\}_{c=1}^{C_0}$ is initialized and orthogonalized by the Gram-Schmidt procedure to obtain $\{\mathbf{FIL}'_c\}_{c=1}^{C_0}$. The orthogonal filters encourage different channels to capture complementary spectral components rather than repeatedly representing the same information. For output channel c, the orthogonal spectral response is

$$f_{\text{ortho}}(\mathbf{F}_{\text{spe}})_c = \sum_{h=1}^{H_f} \sum_{w=1}^{W_f} \mathbf{O}_{h,w,c}\, \mathbf{F}_{\text{spe}}(h, w, c). \tag{12}$$

where H_f and W_f are the spatial dimensions of the feature map, and $\mathbf{O}_{h,w,c}$ denotes the orthogonal filter coefficient.

An excitation module then estimates channel attention weights, which are applied to the spectral feature map. Finally, three cascaded linear layers reconstruct the background image $\hat{\mathbf{X}}_t$. More details of LCSF, LSG, RSDC, and SCE are provided in Appendix A.3.

3.3 Continual Anomaly Detection and Loss Optimization

Loss Function Design. The loss function of CL-SSMamba consists of reconstruction loss L_{recon}, ensemble cooperation loss L_{EC} for K learners, and continual learning loss L_{CL}. The reconstruction loss constrains the fused background estimation of each task:

$$L_{\text{recon}} = \|\hat{\mathbf{X}}_t - \mathbf{X}_t\|^2. \tag{13}$$

To promote cooperation among the continual learners, CL-SSMamba constrains the discrepancy between their normalized background reconstruction responses. The ensemble cooperation loss is formulated as

$$L_{\text{EC}} = \frac{1}{K} \sum_{i=1}^{K} D_{\text{KL}}(P_i \| P_j) = \frac{1}{KN_t} \sum_{k=1, j\neq k}^{K} \sum_{n=1}^{N_t} P_k(X_t^n) \log \frac{P_k(X_t^n)}{P_j(X_t^n)} \tag{14}$$

To balance the plasticity for learning new tasks and the stability for preserving previously acquired knowledge, the continual learning loss is

$$L_{CL}(\boldsymbol{\theta}) = \sum_i I_{1:t-1,i} \left(\boldsymbol{\theta}_i - \boldsymbol{\theta}^*_{1:t-1,i}\right)^2, \tag{15}$$

where $\boldsymbol{\theta}^*_{1:t-1,i}$ denotes the retained optimal value of the i-th parameter after previous tasks and $I_{1:t-1,i}$ represents its importance to previously acquired knowledge. The overall objective function of CL-SSMamba is formulated as:

$$L(\boldsymbol{\theta}) = L_{\text{recon}} + \lambda L_{CL}(\boldsymbol{\theta}) + L_{EC}(\boldsymbol{\theta}). \tag{16}$$

Calculation of Abnormal Scores. During inference, the sequential HSI is fed into the trained CL-SSMamba to obtain the corresponding background reconstruction. The RX detector (Chang & Chiang, 2002) is then applied to the reconstruction residual $\hat{\mathbf{X}}_t - \mathbf{X}_t$ to quantify the final anomaly scores:

$$\mathbf{A}_{\text{map},t} = ((\hat{\mathbf{X}}_t - \mathbf{X}_t) - \boldsymbol{\mu}_m)^T \mathbf{D}^{-1} ((\hat{\mathbf{X}}_t - \mathbf{X}_t) - \boldsymbol{\mu}_m), \tag{17}$$

where $\mathbf{D}^{-1}$ denotes the inverse covariance matrix and $\boldsymbol{\mu}_m$ represents the mean vector of the residuals. The complete training/testing pseudocode and additional formula details are included in Appendix A.3.8.

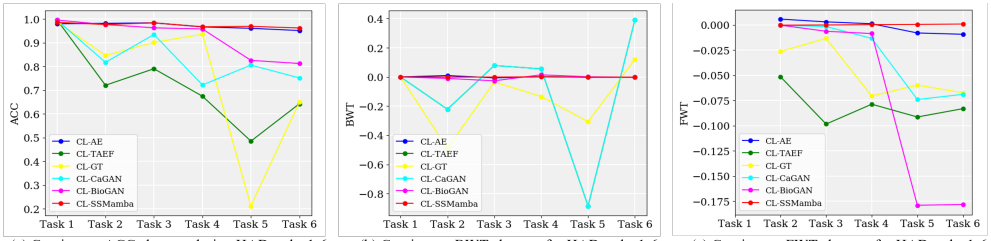
4 Experiments

4.1 Experimental Setting

All experiments were conducted on a computing platform equipped with an Intel Xeon CPU operating at 2.30 GHz, 64 GB of RAM, and an NVIDIA GeForce RTX 3090 GPU with 24 GB of memory. The proposed method was implemented using PyTorch 2.1.1 under Ubuntu. The learning

Table 1: Final-stage continual HAD results on the ABU and HAD100 sequential settings.

Dataset	Method	ACC↑	BWT↑	FWT↑	Avg. AUC_{BS} ↑
ABU (6 tasks)	CL-TAEF	0.6492	0.3917	-0.0832	0.3799
	CL-GT	0.6425	0.1197	-0.0675	0.4682
	CL-CaGAN	0.7514	-0.0801	-0.0690	0.5672
	CL-BioGAN	0.8123	-0.0018	-0.1783	0.7396
	CL-SSMamba	**0.9620**	-0.0012	**0.0008**	**0.9124**
HAD100 (final stage)	CL-TAEF	0.5618	0.0029	-0.4395	0.2076
	CL-GT	0.8057	0.1189	-0.0095	0.5973
	CL-CaGAN	0.9595	0.0101	0.0059	0.8695
	CL-BioGAN	**0.9885**	0.0001	**0.0062**	**0.9332**
	CL-SSMamba	0.9827	-0.0012	0.0013	0.8817



(a) Continuous ACC changes during HAD tasks 1-6 (b) Continuous BWT changes for HAD tasks 1-6 (c) Continuous FWT changes for HAD tasks 1-6

Figure 3: Performance evaluation of ACC, BWT, and FWT on continual HAD tasks.

rate was set to 5×10^{-5}, the sliding-window size was fixed at 3, and the optimal number of continual learners was set to $K = 2$. We evaluate CL-SSMamba on the ABU sequential setting and HAD100, and compare it with CL-TAEF, CL-GT, CL-CaGAN, and CL-BioGAN. Detection performance is evaluated with $\mathrm{AUC}_{(D,F)}$ and AUC_{BS}, while continual behavior is evaluated with Average Accuracy (ACC), Backward Transfer (BWT), and Forward Transfer (FWT). Full dataset descriptions, evaluation definitions, and implementation details are reported in Appendix A.4.1.

4.2 Experimental Results and Analysis

Table 1 reports the final continual-learning results. On the six-task ABU sequence, CL-SSMamba achieves an ACC of 0.9620 and AUC_{BS} of 0.9124, while BWT remains close to zero at -0.0012. Compared with CL-TAEF, CL-GT, CL-CaGAN, and CL-BioGAN, the proposed method obtains the highest final ACC and average AUC_{BS} on ABU. On HAD100, CL-SSMamba achieves ACC 0.9827 and BWT -0.0012, remaining competitive under temporal variation. These results demonstrate that CL-SSMamba can effectively mitigate catastrophic forgetting while maintaining robust background reconstruction across sequential scenes. The performance evaluation of ACC, BWT, and FWT on sequential ABU tasks is illustrated in Fig. 3. CL-SSMamba maintains high ACC during continuous learning and keeps BWT close to zero. The results indicate that the proposed continual-learning loss effectively preserves historical knowledge while the model maintains adaptability to newly arriving tasks. The qualitative results in Fig. 4 further show that CL-SSMamba effectively highlights anomalous regions and achieves high-quality background reconstruction with minimal noise interference, thereby effectively discriminating between anomalies and background.

4.3 Ablation Experiment

We conduct ablation studies on SCE, SLIC, LCSF, LSM, GSM, and the spatial–spectral connection strategy. As shown in Table 2, removing GSM causes the largest performance drop, highlighting the importance of global spatial modeling. Removing SCE, SLIC, or LCSF also degrades performance, confirming their complementary roles. Overall, the full model achieves the highest average AUC_{BS} and the best balance among detection accuracy, background suppression, and continual stability. We

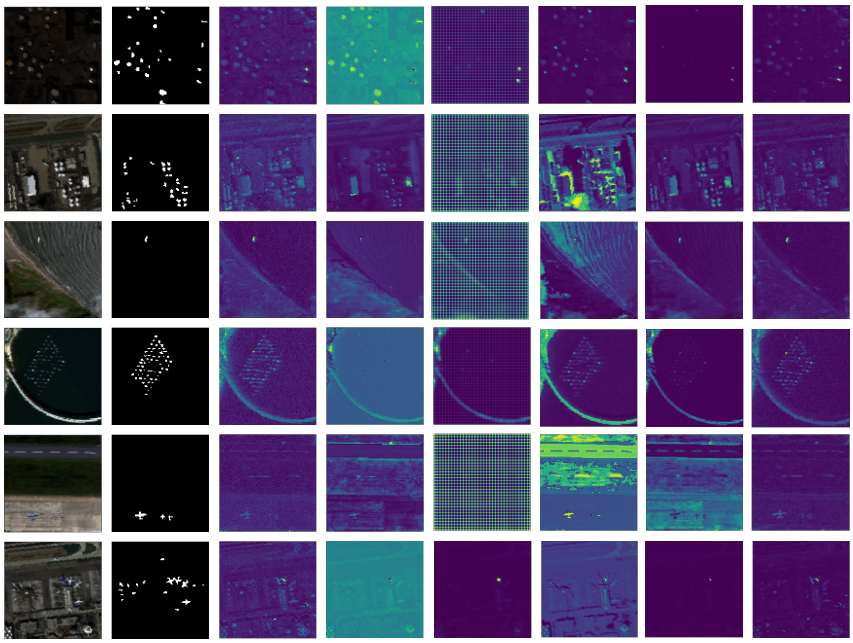


Figure 4: Visualization results of continual HAD over six ABU datasets. From top to bottom: Los Angeles-1, Los Angeles-2, Bay Champagne, San Diego, Gulfport, and Los Angeles-3. (a) ABU pseudo-color image; (b) ground truth annotation; (c) CL-AE; (d) CL-TAEF; (e) CL-GT; (f) CL-CaGAN; (g) CL-BioGAN; (h) CL-SSMamba

Table 2: Compact module ablation on the final ABU stage.

Variant	ACC↑	BWT↑	Avg. AUC_{BS} ↑
w/o SCE	0.9360	-0.0099	0.8931
w/o SLIC	0.9508	-0.0001	0.8975
w/o LCSF	0.9426	-0.0227	0.8743
w/o LSM	**0.9627**	-0.0009	0.9119
w/o SSM	0.9624	**-0.0001**	0.9113
w/o GSM	0.7612	-0.1559	0.7241
Full model	0.9620	-0.0012	**0.9124**

also analyze the number of continual learners. Increasing K from 1 to 2 substantially improves retention, whereas increasing K to 3 does not provide further consistent gains and degrades the six-task setting. Therefore, $K = 2$ is adopted as the default. The complete K-learner results, module-wise trajectories, parameter counts, and computational-efficiency results are provided in Appendix A.4.6.

5 Conclusion

This paper proposes CL-SSMamba, a bio-inspired continual learning framework for open-scenario hyperspectral anomaly detection. Inspired by the compartmentalized memory mechanism of Drosophila, CL-SSMamba employs multiple parallel SSMamba learners to capture complementary spatial-spectral background representations without storing historical HSIs. SLIC and LDPC improve background-anomaly separation, while spatial-spectral Mamba modeling and continual optimization jointly enhance background reconstruction and mitigate catastrophic forgetting. Experiments on sequential HAD tasks demonstrate that CL-SSMamba achieves robust anomaly detection and knowledge retention under continuously changing scenes.

AI USE STATEMENT

Generative AI tools were used for language editing and manuscript organization. All AI-assisted content was reviewed and verified by the authors, who take full responsibility for the final manuscript.

ETHICS STATEMENT

This work uses publicly available hyperspectral remote-sensing datasets and does not involve human subjects or personally identifiable information.

REPRODUCIBILITY STATEMENT

The main paper reports the continual-learning protocol, comparison methods, core hyperparameters, objectives, and principal evaluation metrics. The appendix provides detailed preprocessing equations, module derivations, dataset specifications, metric definitions, full per-task results, and ablation studies. Exact task construction and implementation details are included to support reproducibility.

REFERENCES

Emrecan Bati, Akın Çalışkan, Alper Koz, and A. Aydin Alatan. Hyperspectral anomaly detection method based on auto-encoder. In *Image and Signal Processing for Remote Sensing XXI*, volume 9643, pp. 220–226. SPIE, 2015.

Chein-I Chang and Shao-Shan Chiang. Anomaly detection and classification for hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 40(6):1314–1325, 2002.

Arslan Chaudhry, Marc'Aurelio Ranzato, Marcus Rohrbach, and Mohamed Elhoseiny. Efficient lifelong learning with A-GEM. In *International Conference on Learning Representations*, 2019.

Shuhan Chen, Chein-I Chang, and Xiaorun Li. Component decomposition analysis for hyperspectral anomaly detection. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–22, 2021.

Raphael Cohn, Ianessa Morantte, and Vanessa Ruta. Coordinated and compartmentalized neuromodulation shapes sensory processing in drosophila. *Cell*, 163(7):1742–1755, 2015.

Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Ales Leonardis, Gregory G. Slabaugh, and Tinne Tuytelaars. A continual learning survey: Defying forgetting in classification tasks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(7):3366–3385, 2022. doi: 10.1109/TPAMI.2021.3057446.

Albert Gu and Tri Dao. Mamba: Linear-time sequence modeling with selective state spaces. In *First Conference on Language Modeling*, 2024. URL https://openreview.net/forum?id=tEYskw1VY2.

Qing Guo, Yi Cen, Lifu Zhang, Yan Zhang, and Yixiang Huang. Hyperspectral anomaly detection based on spatial–spectral cross-guided mask autoencoder. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 17:9876–9889, 2024. doi: 10.1109/JSTARS.2024.3393995.

Haoyang He, Yuhu Bai, Jiangning Zhang, Qingdong He, Hongxu Chen, Zhenye Gan, Chengjie Wang, Xiangtai Li, Guanzhong Tian, and Lei Xie. MambaAD: Exploring state space models for multi-class unsupervised anomaly detection. In *Advances in Neural Information Processing Systems*, volume 37, pp. 71162–71187, 2024. doi: 10.52202/079017-2274.

Xu He, Wei An, Yingqian Wang, Qiang Ling, Miao Li, Zaiping Lin, and Shilin Zhou. CWIMamba: Cross-scale windowed integration state space model for hyperspectral anomaly detection. *IEEE Transactions on Geoscience and Remote Sensing*, 63:1–20, 2025. doi: 10.1109/TGRS.2025.3616967.

Lei Hu, Zhiyong Gan, Ling Deng, Jinglin Liang, Lingyu Liang, Shuangping Huang, and Tianshui Chen. ReplayCAD: Generative diffusion replay for continual anomaly detection. In *Proceedings of the Thirty-Fourth International Joint Conference on Artificial Intelligence*, pp. 2946–2954. International Joint Conferences on Artificial Intelligence Organization, 2025. doi: 10.24963/ijcai.2025/328.

Xing Hu, Chun Xie, Zhe Fan, Qianqian Duan, Dawei Zhang, Linhua Jiang, Xian Wei, Danfeng Hong, Guoqiang Li, and Jocelyn Chanussot. Hyperspectral anomaly detection using deep learning: A review. *Remote Sensing*, 14(9):1973, 2022. doi: 10.3390/rs14091973.

Haeyong Kang, Jaehong Yoon, Sung Ju Hwang, and Chang D. Yoo. Continual learning: Forget-free winning subnetworks for video representations. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 47(3):2140–2156, 2025. doi: 10.1109/TPAMI.2024.3518588.

Xudong Kang, Xiangping Zhang, Shutao Li, Kenli Li, Jun Li, and Jón Atli Benediktsson. Hyperspectral anomaly detection with attribute and edge-preserving filters. *IEEE Transactions on Geoscience and Remote Sensing*, 55(10):5600–5611, 2017.

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.

Heesung Kwon and Nasser M. Nasrabadi. Kernel RX-algorithm: A nonlinear anomaly detector for hyperspectral imagery. *IEEE Transactions on Geoscience and Remote Sensing*, 43(2):388–397, 2005.

Jingtao Li, Xinyu Wang, Shaoyu Wang, Hengwei Zhao, and Yanfei Zhong. One-step detection paradigm for hyperspectral anomaly detection via spectral deviation relationship learning. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–15, 2024. doi: 10.1109/TGRS.2024.3392189.

Xiaofan Li, Xin Tan, Zhuo Chen, Zhizhong Zhang, Ruixin Zhang, Rizen Guo, Guanna Jiang, Yulong Chen, Yanyun Qu, Lizhuang Ma, and Yuan Xie. One-for-more: Continual diffusion model for anomaly detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4766–4775, 2025.

Zhaoxu Li, Yingqian Wang, Chao Xiao, Qiang Ling, Zaiping Lin, and Wei An. You only train once: Learning a general anomaly enhancement network with random masks for hyperspectral anomaly detection. *IEEE Transactions on Geoscience and Remote Sensing*, 61:1–18, 2023.

Jie Lian, Lizhi Wang, He Sun, and Hua Huang. GT-HAD: Gated transformer for hyperspectral anomaly detection. *IEEE Transactions on Neural Networks and Learning Systems*, 36(2):3631–3645, 2024.

Haijun Liu, Xi Su, Xiangfei Shen, and Xichuan Zhou. MSNet: Self-supervised multiscale network with enhanced separation training for hyperspectral anomaly detection. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–13, 2024. doi: 10.1109/TGRS.2024.3411009.

Hao Liu and Huaping Liu. Continual learning with recursive gradient optimization. In *International Conference on Learning Representations*, 2022. URL `https://openreview.net/forum?id=7YDLgf9_zgm`.

Zhaojie Pan, Chenyu Li, Antonio Plaza, Jocelyn Chanussot, and Danfeng Hong. Hyperspectral image classification with mamba. *IEEE Transactions on Geoscience and Remote Sensing*, 63:1–14, 2025. doi: 10.1109/TGRS.2024.3521411.

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

Longfei Ren, Lianru Gao, Minghua Wang, Xu Sun, and Jocelyn Chanussot. HADGSM: A unified nonconvex framework for hyperspectral anomaly detection. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–15, 2024. doi: 10.1109/TGRS.2023.3349128.

Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. In *Advances in Neural Information Processing Systems*, volume 30, pp. 2990–2999, 2017.

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*, pp. 11909–11919, 2023. doi: 10.1109/CVPR52729.2023.01146.

Gido M. van de Ven, Hava T. Siegelmann, and Andreas S. Tolias. Brain-inspired replay for continual learning with artificial neural networks. *Nature Communications*, 11(1):4069, 2020. doi: 10.1038/s41467-020-17866-2.

Johannes von Oswald, Christian Henning, Joao Sacramento, and Benjamin F. Grewe. Continual learning with hypernetworks. In *International Conference on Learning Representations*, 2020.

Jianing Wang, Siying Guo, Zheng Hua, Runhu Huang, Jinyu Hu, and Maoguo Gong. CL-CaGAN: Capsule differential adversarial continual learning for cross-domain hyperspectral anomaly detection. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–15, 2024.

Jianing Wang, Zheng Hua, Wan Zhang, Shengjia Hao, Yuqiong Yao, and Maoguo Gong. CL-BioGAN: Biologically-inspired cross-domain continual learning for hyperspectral anomaly detection. *IEEE Transactions on Geoscience and Remote Sensing*, 63:1–15, 2025. doi: 10.1109/TGRS.2025.3561174.

Liyuan Wang, Xingxing Zhang, Qian Li, Mingtian Zhang, Hang Su, Jun Zhu, and Yi Zhong. Incorporating neuro-inspired adaptability for continual learning in artificial intelligence. *Nature Machine Intelligence*, 5(12):1356–1368, 2023a. doi: 10.1038/s42256-023-00747-w.

Shaoyu Wang, Xinyu Wang, Liangpei Zhang, and Yanfei Zhong. Auto-AD: Autonomous hyperspectral anomaly detection network based on fully convolutional autoencoder. *IEEE Transactions on Geoscience and Remote Sensing*, 60:1–14, 2022. doi: 10.1109/TGRS.2021.3057721.

Zhe Wang, Yuting Zhao, and Jun Chen. Hyperspectral anomaly detection based on variational background inference and generative adversarial network. *Pattern Recognition*, 143:109795, 2023b. doi: 10.1016/j.patcog.2023.109795.

Yeming Wen, Dustin Tran, and Jimmy Ba. Batchensemble: An alternative approach to efficient ensemble and lifelong learning. In *International Conference on Learning Representations*, 2020.

Ziyu Wu and Bin Wang. Transformer-based autoencoder framework for nonlinear hyperspectral anomaly detection. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–15, 2024.

Hao-Fang Yan, Yongqiang Zhao, Jonathan Cheung-Wai Chan, and Seong G. Kong. Rapid hyperspectral anomaly detection using discriminative band selection. *IEEE Transactions on Geoscience and Remote Sensing*, 62:1–18, 2024. doi: 10.1109/TGRS.2024.3451559.

A Appendix

A.1 Implementation Details

Following the organization commonly used in ICLR supplementary material, we first provide the experimental and implementation details needed to reproduce the main results. We then define the evaluation metrics, expand the methodological details that were compressed in the main paper, and finally present additional experiments and analyses.

A.1.1 DATASETS AND CONTINUAL TASK CONSTRUCTION

To simulate cross-domain scenarios in a continual learning setting within a continual learning framework, this study employs two publicly available hyperspectral datasets: the ABU dataset Kang et al. (2017) and the HAD100 dataset Li et al. (2023).

The ABU dataset consists of six scenes acquired by the AVIRIS sensor, including Los Angeles-1, Los Angeles-2, Los Angeles-3, Bay Champagne, San Diego and Gulfport, which are organized to construct six sequential HAD tasks. Each scene has a spatial size and characterized by a spatial dimension of 100×100 pixels with 188 to 205 spectral bands covering the wavelength range from 400 nm to 2500 nm, and a spatial resolution ranging from 3.4 m to 7.1 m. The HAD100 dataset comprises five images from the AVIRIS-NG sensor. These images have a spatial size of 64 × 64 pixels and 276 spectral channels observed from 2017 to 2021 with spatial resolutions ranging from 1.9 m to 8.4 m. This dataset is used to evaluate the robustness of the proposed model for continual anomaly detection under temporal variations. Detailed specifications of the two datasets are summarized in Table 3 and Table 4, respectively.

Table 3: Details Of The ABU Anomaly Detection Data Set

HSIs	Spatial size	channels	resolution/m	bands/nm	Sensor
Los Angeles-1	100×100	205	7.1	430 - 860	AVIRIS
Bay Champagne	100×100	188	4.4	400 - 2500	AVIRIS
Los Angeles-2	100×100	205	7.1	430 - 860	AVIRIS
Gulfport	100×100	188	3.4	400 - 2500	AVIRIS
San Diego	100×100	191	7.5	400 - 2500	AVIRIS
Los Angeles-3	100×100	205	7.1	400 - 2500	AVIRIS

Table 4: Details Of The HAD100 Anomaly Detection Data Set

HSIs	Spatial size	channels	resolution/m	Date	Sensor
HAD100_91	64×64	276	2.3	2017/8/21	AVIRIS-NG
HAD100_100	64×64	276	2.3	2017/8/21	AVIRIS-NG
HAD100_24	64×64	276	8.4	2019/10/4	AVIRIS-NG
HAD100_33	64×64	276	1.9	2021/6/14	AVIRIS-NG
HAD100_40	64×64	276	2.0	2017/9/8	AVIRIS-NG

A.1.2 TRAINING CONFIGURATION AND COMPARISON METHODS

Implementation. All experiments were conducted on a computing platform equipped with an Intel Xeon CPU operating at 2.30 GHz, 64 GB of RAM, and an NVIDIA GeForce RTX 3090 GPU with 24 GB of memory.

The proposed method was implemented based on the PyTorch 2.1.1 deep learning framework under the Ubuntu operating system. Regarding hyperparameter configurations, the learning rate was set to 5×10^{-5}, and the sliding window size was fixed at 3. Additionally, subsequent ablation studies demonstrated that the optimal number of continual learners is $K = 2$.

Comparison methods. To evaluate the effectiveness of the proposed method, six representative continual HAD models were selected for comparative analysis, covering traditional Autoencoder, generative models, and bio-inspired architectures. The benchmark methods include: 1) CL-AE and CL-TAEF: Adaptations of the standard AE Bati et al. (2015) and Tensor AE framework Wu & Wang (2024), respectively, both augmented with Elastic Weight Consolidation (EWC) loss to mitigate forgetting; 2) CL-GT: CL-GT is derived from the Graph Transformer model Lian et al. (2024) with

the incorporation of EWC constraints; 3) CL-CaGAN: The pioneering Generative Adversarial Network (GAN) designed specifically for continual HAD tasks Wang et al. (2024); 4) CL-BioGAN: A method based on biologically inspired generative models Wang et al. (2025); and 5) CL-SSMamba: The proposed method in this paper.

A.2 Evaluation Metrics

The evaluation separates anomaly-detection quality from continual-learning behavior. Detection quality is measured by the AUC family and background-suppression score, while ACC, BWT, and FWT quantify performance retention and transfer over the task sequence.

To provide an objective and comprehensive evaluation of continual hyperspectral anomaly detection models, this study adopts a multi-dimensional evaluation framework. Specifically, core metrics are categorized into three key aspects: anomaly detection accuracy, background suppression capability, and continual learning performance.

To quantitatively evaluate the detection performance, this study employs three distinct Area Under the Curve (AUC) metrics derived from 2D Receiver Operating Characteristic (ROC) analysis, according to the characteristics of hyperspectral anomaly detection. These metrics include: $\mathrm{AUC}_{(D,F)}$, which is calculated from Probability of Detection P_D versus Probability of False Alarm P_F curve; $\mathrm{AUC}_{(D,\tau)}$, which is derived from the relationship between P_D and the Decision Threshold τ; and $\mathrm{AUC}_{(F,\tau)}$, which is derived from the curve plotting P_F against τ. Here, P_D denotes the detection probability of anomalous targets, P_F represents the probability of incorrectly classifying background pixels as anomalies, also known as the false alarm rate, and τ signifies the decision threshold. $\mathrm{AUC}_{(D,F)}$ served as the primary metric, where a value approaching 1 indicates superior discriminative capability between anomalies and background. In contrast, a lower $\mathrm{AUC}_{(F,\tau)}$ value indicates better background suppression capability, while $\mathrm{AUC}_{(D,\tau)}$ reflects the detection sensitivity under varied threshold settings.

To provide a comprehensive evaluation of the model's performance in terms of both anomaly detection accuracy and background suppression capability, we employ the Anomaly Detection Accuracy versus Background Suppression, denoted as AUC_{BS} metric Chen et al. (2021). This metric measures the trade-off between detection capability and false alarm suppression by calculating the difference between $\mathrm{AUC}_{(D,F)}$ and $\mathrm{AUC}_{(F,\tau)}$. The mathematical formulation is given by:

$$\mathrm{AUC_{BS}} = \mathrm{AUC}(D,F) - \mathrm{AUC}(F,\tau) \tag{18}$$

A higher AUC_{BS} value indicates that the model achieves a superior trade-off between sensitivity and specificity, simultaneously maximizing the anomaly detection rate while suppressing background derivation. This characteristic effectively mitigates false alarms caused by complex spectral interference, rendering the model highly robust and suitable for practical engineering deployment.

To evaluate the model performance in terms of knowledge retention and transfer capability in sequential learning scenarios, this study adopts three standard metrics: Average Accuracy (ACC), Backward Transfer (BWT), and Forward Transfer (FWT). These metrics are derived from the $\mathrm{AUC}_{(D,F)}$ values recorded during the continual training process. Specifically, ACC measures the overall detection performance after the completion of the entire task sequence. It is defined as the average $\mathrm{AUC}_{(D,F)}$ across all observed tasks, formulated as follows:

$$ACC = \frac{1}{T}\sum_{i=1}^{T} AUC_{(D,F)T,i} \tag{19}$$

where T denotes the total number of tasks, and $\mathrm{AUC}_{(D,F)_{T,i}}$ represents the test performance on task i after the model has finished learning task T.

Backward Transfer (BWT) measures the average degradation in the performance of previously learned tasks after the model has completed the entire sequence of T tasks. The metric is formulated as:

$$BWT = \frac{1}{T-1}\sum_{i=1}^{T-1}(AUC_{(D,F)T,i} - AUC_{(D,F)T-1,i}) \tag{20}$$

where $\mathrm{AUC}_{(D,F)T-1,i}$ denotes the performance on task i after the model has learned the $(T-1)$-th task. The BWT value approaching zero or exhibiting better value indicates superior knowledge retention for historical learned tasks, signifying effective mitigation of catastrophic forgetting.

Forward Transfer (FWT) quantifies the model's capability to leverage knowledge acquired from previous tasks to facilitate the learning of subsequent tasks. The metric is formulated as:

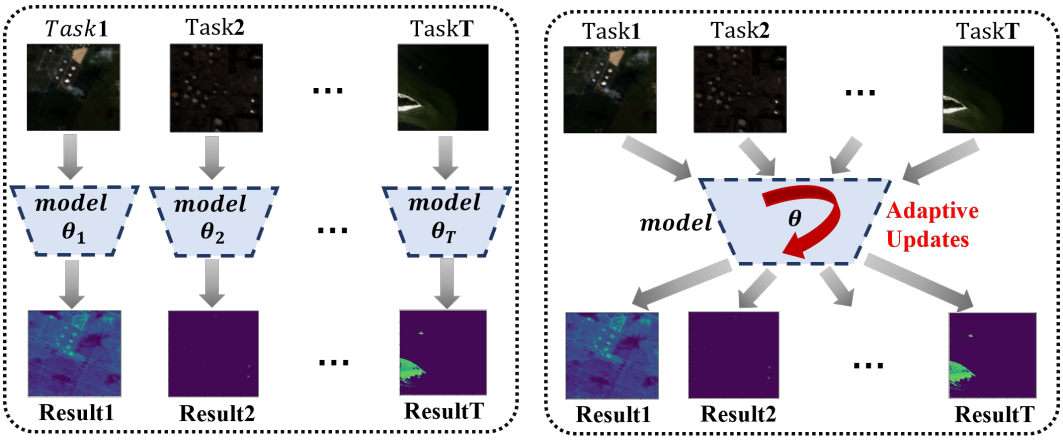
$$FWT = \frac{1}{T-1}\sum_{i=2}^{T}\left(AUC_{(D,F)i,i} - AUC^{*}_{(D,F)i}\right) \tag{21}$$

In this formulation, $\mathrm{AUC}^{*}_{(D,F)i}$ denotes the baseline performance obtained by training the model on the i-th task independently, when the model is trained on the i-th task independently. In contrast, $\mathrm{AUC}_{(D,F)i,i}$ represents the performance achieved on task i within the sequential learning framework. A positive FWT value indicates that prior knowledge can be effectively transferred to new tasks, thereby improving the learning efficiency for incoming tasks.

A.3 Continual HAD Setting and Detailed Methodology

This section expands the continual-HAD formulation and the CL-SSMamba architecture. We first clarify the transition from static HAD to continual HAD and the biological motivation for the parallel learner design. We then provide the full preprocessing, spatial-spectral state-space modeling, continual objective, and inference procedure that are summarized more compactly in the main paper.

A.3.1 From Static HAD to Continual HAD



(a) Hyperspectral Anomaly Detection with Traditional Deep Learning

(b) Hyperspectral Anomaly Detection with Continuous Learning

Figure 5: Schematic Diagram of Hyperspectral Anomaly Detection Using Traditional Deep Learning Methods and Continual Learning Methods.

The comparison above motivates the sequential formulation used in this work: a detector is updated as new hyperspectral scenes arrive, while historical raw HSIs are not retained for replay. The main paper focuses on the resulting stability-plasticity problem; the following subsection explains why the proposed multi-learner design is a natural structural response to that problem.

A.3.2 Biological Inspiration and Algorithmic Mapping

Biological learning systems provide a valuable inspiration for designing memory-efficient continual learning models. In particular, as illustrated in Fig.6, the drosophila mushroom body is a compact neural structure responsible for associative learning and memory formation rather than storing new and old experiences through a single undifferentiated pathway, the mushroom body consists

of multiple parallel and compartmentalized functional regions, such as the γ2-γ5 compartments, which are modulated by distinct dopaminergic neurons. These compartmentalized pathways allow newly acquired information to be rapidly encoded while reducing destructive interference with previously consolidated memories. This biological mechanism suggests an efficient continual learning principle: sequential experiences can be processed by multiple specialized memory pathways, while modulatory signals coordinate their learning dynamics to balance plasticity and stability. Such multi-compartment sequential processing offers an efficient and low-energy solution for continual learning. Meanwhile, neural sequence models based on selective state-space architectures (Mamba) have recently shown strong potential for modeling long-range dependencies with efficient computation. Motivated by this biological insight, their integration into hyperspectral continual anomaly detection has not been explored by now.

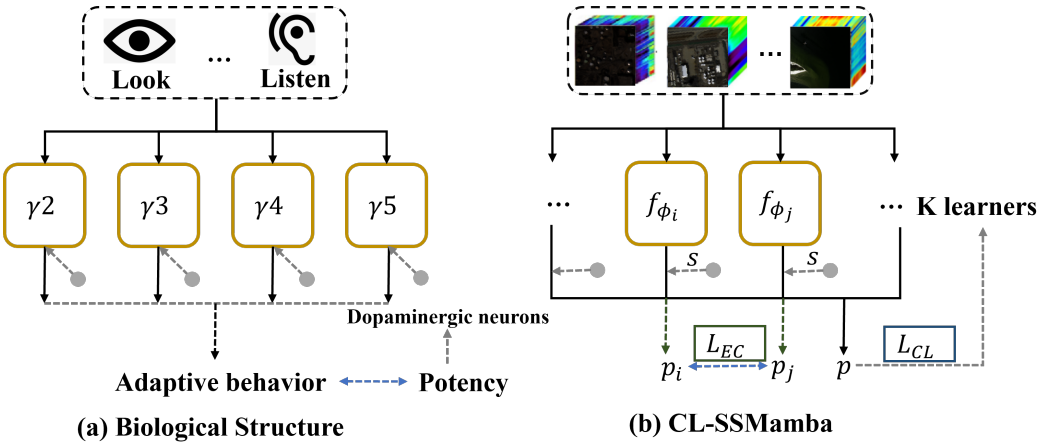


Figure 6: (a) Drosophila learn and memorize sequentially through multiple parallel compartments (subnetworks, γ2-5) modulated by dopaminergic neurons. (b) Inspired by biological learning systems, a proposed strategy for cooperative learning among multiple sequential learners.

Inspired by the parallel processing mechanisms of biological systems, in this paper, we develop a collaborative multi-learner paradigm to achieve efficient continual learning and high-fidelity background reconstruction. A continual background reconstruction learning network based on autoencoder (AE) via neural subnetwork fusion in a spatial-spectral Mamba (CL-SSMamba) is designed to realize robust Hyperspectral Anomaly Detection task for open-scenario circumstance. A global spatial feature extraction Mamba module is designed to enhance the understanding of spatial structure information and improve the ability to distinguish abnormal targets. Meanwhile, a local spectral feature extraction Mamba module is proposed to capture the correlation of different spectral bands through orthogonal filters for improving the spectral distinguishability. Furthermore, inspired by the parallel processing mechanisms of biological systems, a collaborative multi-learner paradigm is elaborately designed to achieve efficient continual learning and high-fidelity background reconstruction.

The proposed CL-SSMamba translates the compartmentalized organization of the drosophila mushroom body into a collaborative multi-learner architecture. Specifically, instead of employing a single AE-Mamba network to encode all sequential hyperspectral domains, CL-SSMamba constructs K parallel SSMamba subnetworks, where in subnetwork learner θ each learner can be regarded as an independent memory compartment for modeling complementary background representations with sequential Mamba learning module from OHAD tasks. The outputs of these learners are then aggregated to generate a stable background reconstruction, while the ensemble cooperation loss (L_{EC}) encourages consistency among different learners and the continual learning loss (L_{CL}) constrains the model to preserve previously acquired knowledge. In this way, the proposed framework transforms the biological mechanism of dopaminergic modulation and compartmentalized memory into an algorithmic form of multi-subnetwork collaboration and knowledge retention.

Different from conventional replay-based continual learning methods, the proposed CL-SSMamba does not rely on retaining raw historical hyperspectral data for replay. Instead, previous-domain knowledge is preserved through the parameters of multiple SSMamba subnetworks and constrained through collaborative and continual-learning loss terms. Therefore, the proposed biologically inspired architecture reduce memory overhead while improving the stability of background reconstruction. Moreover, by incorporating selective state-space modeling, each SSMamba learner can efficiently capture long-range spatial-spectral dependencies, further enhancing anomaly discrimination in complex hyperspectral backgrounds.

A.3.3 Background-Aware Preprocessing: SLIC and LDPC

The data preprocessing module contains two consecutive stages rather than two independent clustering operations. In the first stage, Simple Linear Iterative Clustering (SLIC) partitions the HSI into spatially compact and spectrally homogeneous superpixels. The purpose of this stage is not to directly identify anomalous pixels. Instead, it increases the local proportion of a spatially coherent but globally sparse background category within an individual superpixel. This local enhancement makes the sparse background more distinguishable from isolated anomalies.

In the second stage, Local Density Peak Clustering (LDPC) is applied to the superpixel-level representations generated by SLIC. LDPC evaluates the local density and the relative distance of each superpixel. Superpixels associated with dense and repeatedly occurring structures are assigned to background clusters, whereas isolated or low-density superpixels are treated as anomaly candidates. Therefore, SLIC establishes reliable local regions, and LDPC subsequently performs background-anomaly separation at the region level.

Taking the 8×8 schematic patches in Fig.7 as an example, the anomalous pixels (gray-purple) and sparse-background pixels (light-green) only respectively occupy 6.25% and 9.375% proportions in the whole patch, representing extremely low proportion in whole patches. Traditional direct clustering algorithm may confuse these two categories, while after SLIC partitioning, the spatially coherent sparse-background pixels are concentrated within a local superpixel to increase the local proportion. In contrast, anomalous pixels remain spatially isolated to provides a more reliable basis for the subsequent density-peak analysis.

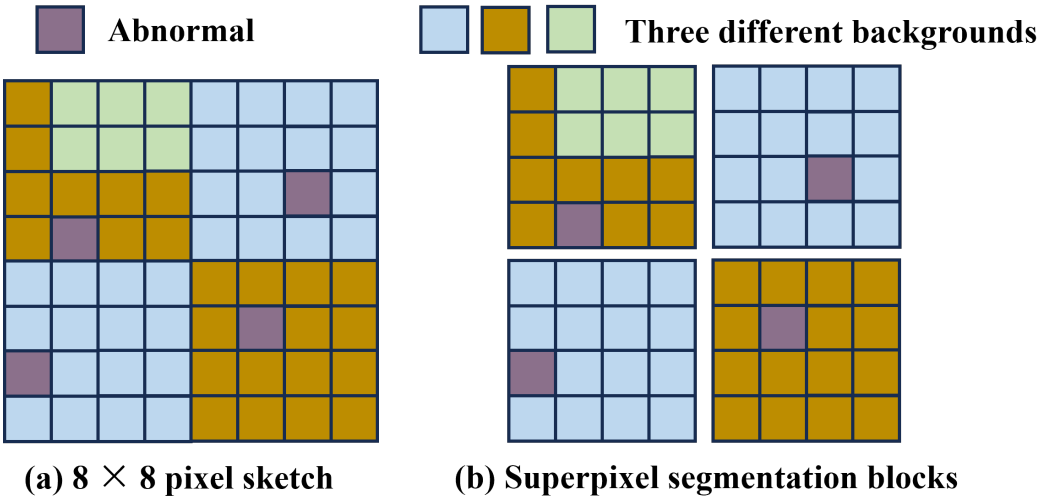


Figure 7: Superpixel Segmentation Block Diagram.

Suppose the current task t, where $t = 1, \ldots, T$, and T is the total number of continuous leaning tasks. For each input HSI $\mathbf{X}_t \in \mathbb{R}^{H \times W \times C}$ is divided into Z patches with spatial size $N \times N$, where H and W respectively denote the height and width of $\mathbf{X}_t$. Therefor the z-th patch of task t is denoted as $\mathbf{X}_t^z$, where $z = 1, \ldots, Z$. For spatial initialization, each index of $N \times N$ regular patch, can be represented as $m = 1, \ldots, T$ denotes the index of a grid region, and a_m and b_m respectively

denote its row and column indices. The initial spatial center of region m is defined as

$$\mathbf{c}_{t,m}^{(0)} = \left(\left\lfloor \frac{(2a_m + 1)H}{2N} \right\rfloor, \left\lfloor \frac{(2b_m + 1)W}{2N} \right\rfloor \right), \quad m = 1, \ldots, N^2. \tag{22}$$

At SLIC iteration v, let $\boldsymbol{\mu}_m^{(v)}$ and $\mathbf{c}_m^{(v)}$ respectively denote the spectral centroid and spatial center of superpixel m. For each pixel, $\mathbf{x}_i$ is its spectral vector and $\mathbf{p}_i$ its two-dimensional spatial coordinate. The composite distance between pixel i and superpixel m is

$$d_{i,m}^{(v)} = \left\| \mathbf{x}_i - \boldsymbol{\mu}_m^{(v)} \right\|^2 + \lambda_{\text{sp}} \left\| \mathbf{p}_i - \mathbf{c}_m^{(v)} \right\|^2 \tag{23}$$

where λ_{sp} is the spatial-distance weighting factor typically set to 8×10^{-3}. Each pixel is assigned to the superpixel center with the minimum composite distance.

Let $\mathcal{S}_m^{(v)}$ denote the pixel set assigned to superpixel m at iteration v. After calculating the mean spectrum of this set, the pixel whose spectrum is closest to the mean spectrum is selected as the updated spatial center:

$$\mathbf{c}_m^{(v+1)} = \underset{i \in \mathcal{S}_m^{(v)}}{\arg\min} \left\| \mathbf{x}_i - \frac{1}{\left| \mathcal{S}_m^{(v)} \right|} \sum_{j \in \mathcal{S}_m^{(v)}} \mathbf{x}_j \right\|^2 \tag{24}$$

the spatial center iteration proceeds until the cluster centers cease to update, signifying that the algorithm has converged.

After SLIC convergence, each superpixel is represented by its mean spectral vector. LDPC then calculates its local density and its minimum distance to a higher-density superpixel. Superpixels with simultaneously high local density and large relative distance are selected as density peaks. The remaining superpixels are assigned to their nearest higher-density neighbors. Large and dense clusters are interpreted as reliable background structures, whereas small or isolated clusters are regarded as potential anomalies. The prepossessing module finally produces the optimized tensor dataset for task t represented as $\bar{\mathbf{X}}_t$for overlapping-patch generation and subsequent background reconstruction.

A.3.4 Collaborative Multi-Learner Architecture

The SSMamba mainly consists of K parallel learners and each learner are sequentially comprised of global spatial Mamba and spectral Mamba.

Multiple Parallel SSMamba Learners Inspired by the biological learning mechanism of Drosophila, the proposed CL-SSMamba departs from the conventional single-learner paradigm and employs K parallel learners based on multiple parallel compartments process sequential information under dopaminergic regulation Cohn et al. (2015). The k-th learner is implemented by the SSMamba subnetwork $f_{\theta_k}(\cdot)$, and each learner is instantiated based on the proposed SSMamba subnetwork $f_{\theta_k}(\cdot)$, operating within a unified parameter space to execute sequential tasks for open scenario circumstance.

Considering the unlabeled feature of continual HAD tasks, a weighting parameter $s = 1/T$, where T denotes the total number of tasks, is introduced to the background reconstruction output of each sequential learner to effectively merge the task-specific information. The final background image output formula is as follows:

$$\hat{\mathbf{X}}_t = \sum_{i=1}^{K} s \cdot f_{\theta_k}(\bar{\mathbf{X}}_t) \tag{25}$$

where $\bar{\mathbf{X}}_t$ denotes the preprocessed input of task t. The complete set of trainable learner parameters is

$$\boldsymbol{\theta}_k^t = \{\boldsymbol{\phi}_{spa}^t, \boldsymbol{\phi}_{spe}^t, \boldsymbol{\phi}_{sce}^t, \boldsymbol{\phi}_{rec}^t\} \tag{26}$$

where $\boldsymbol{\phi}_{spa}^t, \boldsymbol{\phi}_{spe}^t, \boldsymbol{\phi}_{sce}^t$ and $\boldsymbol{\phi}_{rec}^t$ respectively denote the sequential Mamba learning parameters for Global Spatial Mamba Module (GSM) , Local Spectral Mamba Module (LSM), Spectral Correlation Enhancement Module (SCE), and the reconstruction head for the current t-th learning HAD task.

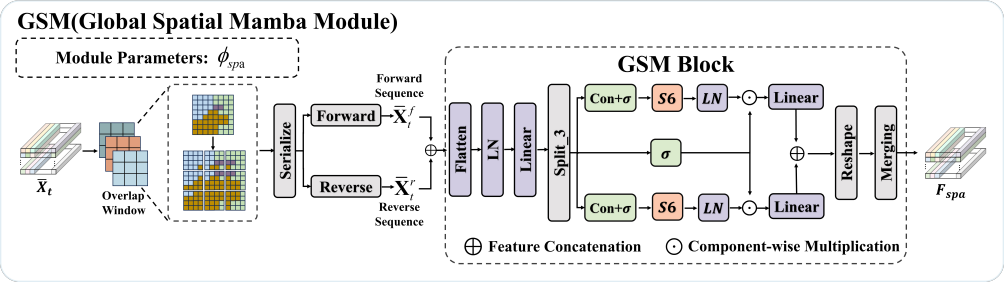


Figure 8: The detail structure of the proposed Global Space Mamba (GSM) feature extraction module.

A.3.5 Global Spatial Mamba Modeling

In view of Mamba can model long-range dependencies in linear complexity, which can effectively mitigate the inherent feature extraction limitations of CNNs and reduce computation cost in Transformers. The proposed GSM Feature Extraction module is designed to capture global spatial structural information within the image, the detailed architecture of GSM module is illustrated in Fig.8. To mitigate the disruption of local neighborhood pixel correlations and the inadequacy of spatial feature representation inherent in traditional patch generation, we employ overlapping window strategy. The stride of the sliding window is set to half the patch size to ensure spatial continuity. Subsequently, the processed tensor dataset $\bar{\mathbf{X}}_t$ is serialized into a forward sequence $\bar{\mathbf{X}}_t^f$ and a reverse sequence $\bar{\mathbf{X}}_t^r$, both of which are fed into the Mamba block for bidirectional modeling.

Global spatial dependencies were modeled using layer normalization (LN), linear transformation (Linear), one-dimensional convolution (Conv), the SiLU activation function (σ), and the S6 state space operator. The key equations are represented as follows.

$$\tilde{\mathbf{X}}_t^f, \tilde{\mathbf{X}}_t^r, \tilde{\mathbf{X}}_t^z = \mathrm{Split}_3\left[\mathrm{Linear}\left(\mathrm{LN}\left(\mathrm{Flatten}\left(\bar{X}_t^f \oplus \bar{X}_t^r\right)\right)\right)\right] \tag{27}$$

where $\oplus$ denotes feature concatenation, and $\mathrm{Split}_3(\cdot)$ denotes an equal partition of the projected feature along the channel dimension into three sub-features. Specifically, $\tilde{\mathbf{X}}_t^f$ and $\tilde{\mathbf{X}}_t^r$ are assigned to the forward and reverse state-space branches, respectively, while $\tilde{\mathbf{X}}_t^z$ serves as the gating feature for subsequent feature modulation. The forward and reverse state-space representations are then computed as follows.

$$\tilde{\mathbf{X}}_t^{f'} = \mathrm{S6}\left[\sigma\left(\mathrm{Conv}\left(\tilde{\mathbf{X}}_t^f\right)\right)\right] \tag{28}$$

$$\tilde{\mathbf{X}}_t^{r'} = \mathrm{S6}\left[\sigma\left(\mathrm{Conv}\left(\tilde{\mathbf{X}}_t^r\right)\right)\right] \tag{29}$$

The forward and reverse state-space representations are subsequently modulated by the gating feature $\tilde{\mathbf{X}}_t^z$. Here, $\odot$ denotes component-wise multiplication, through which the gating feature scales the corresponding feature responses. The modulated features are then linearly projected and combined with the corresponding residual inputs as follows, where $\tilde{\mathbf{F}}_t^f$ and $\tilde{\mathbf{F}}_t^r$ respectively denote the resulting feature representations of the forward and reverse branches.

$$\tilde{\mathbf{F}}_t^f = \mathrm{Linear}\left[\mathrm{LN}\left(\tilde{\mathbf{X}}_t^{f'}\right) \odot \sigma\left(\tilde{\mathbf{X}}_t^z\right)\right] + \mathrm{Flatten}\left(\bar{\mathbf{X}}_t^f\right) \tag{30}$$

$$\tilde{\mathbf{F}}_t^r = \mathrm{Linear}\left[\mathrm{LN}\left(\tilde{\mathbf{X}}_t^{r'}\right) \odot \sigma\left(\tilde{\mathbf{X}}_t^z\right)\right] + \mathrm{Flatten}\left(\bar{\mathbf{X}}_t^r\right) \tag{31}$$

The bidirectional outputs are fused and reshaped as

$$\tilde{\mathbf{F}}_t^o = \mathrm{Reshape}\left(\tilde{\mathbf{F}}_t^f \oplus \tilde{\mathbf{F}}_t^r\right) \tag{32}$$

through feature fusion and patch reconstruction, global spatial feature output can be represented as

$$\mathbf{F}_{\mathrm{spa}} = \mathrm{Merging}\left(\tilde{\mathbf{F}}_t^o\right) \tag{33}$$

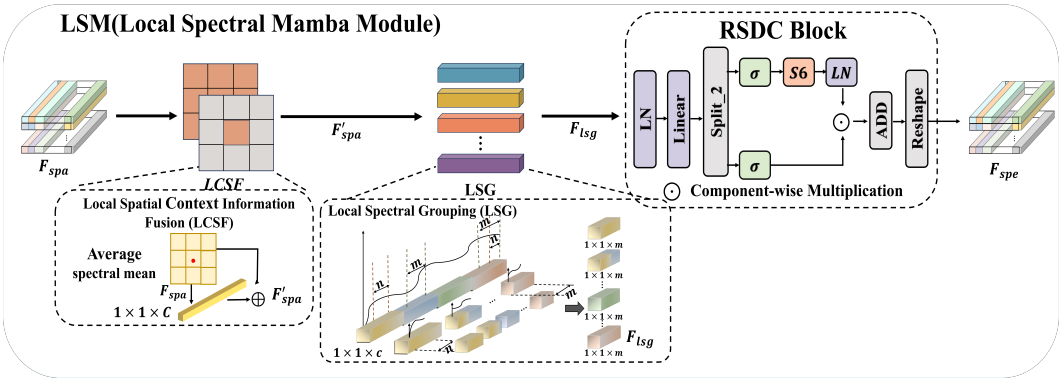


Figure 9: The detail structure of Local Spectral Mamba (LSM) feature extraction module.

where $\mathrm{Merging}(\cdot)$ denotes the spatial reassembly of the processed patch-level representations according to their original positions in the input HSI. Specifically, the outputs of all patches are arranged back into their corresponding spatial locations and aggregated to reconstruct the complete global spatial feature map $\mathbf{F}_{\mathrm{spa}}$.

The bidirectional modeling operation captures global spatial dependencies, whereas parameter sharing between the two directions limits additional computational overhead and improves robustness to spatial noise.

A.3.6 Local Spectral Mamba Modeling

The local spectral branch progressively refines the representation through local context fusion, spectral grouping, long-range spectral state-space modeling, and correlation enhancement. The following paragraphs detail these operations in their execution order.

Local spectral features are instrumental in capturing subtle variations provide broader contextual information, facilitating the differentiation of spectral characteristics across diverse samples. To leverage the complementary advantages of these features, distinct from GSM, the LSM module is specifically engineered for the joint extraction of spectral and local spatial features, the detailed architecture of which is depicted in Fig.9. Focusing on the joint extraction of spectral and local spatial features, the process involves three key steps.

Local Spatial Context Fusion (LCSF). Let (i, j) denote a spatial position, LCSF calculates the local mean $m_{i,j}$ of pixels using a $w \times w$ sliding window as follows:

$$\mathbf{m}_{i,j} = \frac{1}{w^2} \sum_{i'=i-w/2}^{i+w/2} \sum_{j'=j-w/2}^{j+w/2} \mathbf{F}_{i',j'} \tag{34}$$

fused with the original pixel features to obtain

$$\mathbf{f}'_{i,j} = \mathbf{f}_{i,j} + \mathbf{m}_{i,j} \tag{35}$$

and subsequently generate $\mathbf{F}'_{\mathrm{spa}}$.

Local Spectral Grouping (LSG). LSG groups the $\mathbf{F}'_{\mathrm{spa}}$ spectrum according to the following formula to reduce redundancy:

$$g = \lfloor \frac{C - (m-n)}{n} \rfloor + \lceil \frac{(c - (m-n)) \bmod n + m - n}{n} \rceil \tag{36}$$

where C denotes the number of channels, m and n respectively denote the spectral-group width and group stride, and outputs $\mathbf{F}_{\mathrm{lsg}}$.

Remote Spectral Dynamic Correlation (RSDC). The RSDC module is employed to model long-range dependencies among spectral groups. Specifically, $\mathbf{F}_{\text{lsg}}$ is first normalized and linearly projected, and then divided into the main spectral branch $\mathbf{F}_s$ and the gating branch $\mathbf{F}_z$ as

$$\mathbf{F}_s, \mathbf{F}_z = \text{Split}_2\left(\text{linear}\left(\text{LN}(\mathbf{F}_{\text{lsg}})\right)\right) \tag{37}$$

where $\text{Split}_2(\cdot)$ partitions the projected feature into two sub-features along the channel dimension. The main spectral feature is subsequently processed by convolution, SiLU activation function (σ), and the S6 operator.

$$\mathbf{F}_{S6} = S6\left(\sigma(\mathbf{F}_s)\right) \tag{38}$$

The state-space representation is then modulated by the gating feature and combined with the residual input to obtain the spectral representation:

$$\mathbf{F}_{\text{spe}} = \text{Reshape}(\text{linear}(\text{LN}(\mathbf{F}_{S6}) \odot \text{Silu}(\mathbf{F}_z)) + \mathbf{F}_{\text{lsg}}) \tag{39}$$

Here, $\mathbf{F}_{S6}$ denotes the feature generated by the S6 operator, and $\mathbf{F}_{spe}$ represents the output spectral feature of RSDC. Let $\mathbf{F}_s = [\mathbf{f}_1, \mathbf{f}_2, \ldots, \mathbf{f}_g]^T$ denote the spectral feature sequence fed into the S6 operator, where g is the number of spectral groups. $\mathbf{f}p$ denotes the feature vector at the p-th spectral position, with $p = 1, \ldots, g$. The S6 operation on $\mathbf{F}s$ is formulated as

$$\mathbf{h}_p = \overline{\mathbf{A}} \odot \mathbf{h}_{p-1} + \mathbf{B}(\Delta \odot \mathbf{f}_p) \tag{40}$$

$$\mathbf{f}_p^{'} = \mathbf{C}\mathbf{h}_p + \mathbf{D} \odot \mathbf{f}_p \tag{41}$$

where $\mathbf{h}_p$ and $\mathbf{h}_{p-1}$ respectively denote the current and previous hidden states, $\overline{\mathbf{A}}$ denotes the discretized state transition parameter. The parameters $\mathbf{B}, \mathbf{C}$, and $\mathbf{\Delta}$ are determined by the input sequence $\mathbf{F}_s$. $\mathbf{D}$ denotes the skip-connection parameter.

As indicated by Eq(40) and Eq(41), the spectral correlation information is propagated along the sequence through the recurrent update of the state (h). Specifically, the hidden state at each spectral position integrates both the previous state information and the current spectral input. In this manner, RSDC models long-range spectral dependencies in accordance with the sequential characteristics of hyperspectral data, thereby improving the representation capability for cross-band spectral correlations.

Spectral Correlation Enhancement Module (SCE). Adjacent hyperspectral bands are highly correlated and frequently contain redundant information. To reduce this redundancy, the filter set $\{\mathbf{FIL}_c\}_{c=1}^{C_0}$ is initialized and orthogonalized by the Gram-Schmidt procedure to obtain $\{\mathbf{FIL}_c'\}_{c=1}^{C_0}$. The orthogonal filters encourage different channels to capture complementary spectral components rather than repeatedly representing the same information.

For output channel c, the orthogonal spectral response is

$$f_{\text{ortho}}\left(\mathbf{F}_{\text{spe}}\right)_c = \sum_{h=1}^{H_f} \sum_{w=1}^{W_f} \mathbf{O}_{h,w,c}\, \mathbf{F}_{\text{spe}}(h, w, c). \tag{42}$$

Here, H_f and W_f are the spatial dimensions of the feature map, and $\mathbf{O}_{h,w,c}$ denotes the orthogonal filter coefficient. The resulting global spectral feature is represented by $\mathbf{S} \in \mathbb{R}^{1\times1\times C_0}$. An excitation module then estimates channel attention weights, which are applied to the spectral feature map. Orthogonalization reduces repeated spectral responses and consequently decreases the feature redundancy, storage demand, and computational burden of the subsequent enhancement and reconstruction layers. Finally, three cascaded linear layers reconstruct the background image $\widehat{\mathbf{X}}_t$.

A.3.7 Continual Objective and Knowledge Retention

The continual objective is designed to couple three complementary requirements: accurate background reconstruction, agreement among parallel learners, and retention of previously acquired knowledge. The complete objective omitted from the main text is given below.

The loss function for the proposed CL-SSMamba consists of reconstruction loss L_{recon}, ensemble cooperation loss L_{EC} for K learners and the continual learning loss L_{CL}. The reconstruction loss is first employed to constrain the fused background estimation of each task. For task t, it is defined as

$$L_{\text{recon}} = \|\hat{\mathbf{X}}_t - \mathbf{X}_t\|^2 \tag{43}$$

To promote cooperation among the continual learners, CL-SSMamba constrains the discrepancy between their background reconstruction responses. For the n-th sample of task t, let $\widehat{\mathbf{X}}_t^{n(k)}$ denote the reconstruction output generated by the k-th learner. Its normalized reconstruction response distribution is defined as

$$\mathbf{P}_k\left(X_t^n\right) = \text{Softmax}\left(\widehat{\mathbf{X}}_t^{n(k)}\right) \tag{44}$$

where $k = 1, \ldots, K$, and K denotes the number of parallel learners. $\mathbf{P}_k(X_t^n)$ therefore represents the normalized reconstruction response distribution produced by the k-th learner for the same input sample.

Based on these normalized reconstruction distributions, the Kullback-Leibler (KL) divergence is employed to measure the discrepancy between different learners. Accordingly, the Ensemble Cooperation (EC) loss is formulated as

$$L_{\text{EC}} = \frac{1}{K}\sum_{i=1}^{K} D_{\text{KL}}(P_i \| P_j) = \frac{1}{KN_t}\sum_{k=1, j\neq k}^{K}\sum_{n=1}^{N_t} P_k(X_t^n)\log\frac{P_k(X_t^n)}{P_j(X_t^n)} \tag{45}$$

where N_t denotes the number of samples involved in task t, and X_t^n denotes the n-th sample of task t. $P_k(X_t^n)$ and $P_j(X_t^n)$ represent the normalized reconstruction response distributions produced by the k-th and j-th learners for the same input sample, respectively. The EC loss therefore constrains the distribution discrepancy between different learners and encourages consistent background reconstruction.

To balance the plasticity for learning new tasks and the stability for preserving previously acquired knowledge, the continual learning loss is formulated as

$$L_{CL}(\boldsymbol{\theta}) = \sum_i I_{1:t-1,i}\left(\boldsymbol{\theta}_i - \boldsymbol{\theta}^*_{1:t-1,i}\right)^2 \tag{46}$$

where $\boldsymbol{\theta}^*_{1:t-1,i}$ denotes the optimal value of the i-th model parameter retained after learning the previous tasks, and $I_{1:t-1,i}$ represents the importance of this parameter to the previously acquired knowledge. By penalizing changes in important parameters, the continual learning loss mitigates catastrophic forgetting while maintaining adaptability to the current task.

By combining Eq(43), Eq(45) and Eq(46), the overall objective function of CL-SSMamba is formulated as:

$$L(\boldsymbol{\theta}) = L_{\text{recon}} + \lambda L_{\text{CL}}(\boldsymbol{\theta}) + L_{\text{EC}}(\boldsymbol{\theta}) \tag{47}$$

where the coefficient λ controls the strength of the historical knowledge and the adaptation for current task. The detail pseudo-code for the proposed CL-SSMamba is introduced in Algorithm 1.

A.3.8 Anomaly Scoring and Full Training/Inference Procedure

At inference time, the reconstruction residual is converted into an anomaly map using the RX detector. We then summarize the full continual training and testing procedure to connect the preprocessing, multi-learner backbone, loss optimization, and anomaly scoring steps into a single pipeline.

Anomaly scoring. During the inference phase, the sequential data stream $\mathbf{X}_1, \ldots, \mathbf{X}_T$ is fed into the trained CL-SSMamba model to yield the corresponding background reconstructions $\hat{\mathbf{X}}_1, \ldots, \hat{\mathbf{X}}_T$. Subsequently, the RX detector is applied to the computed reconstruction residuals $(\hat{\mathbf{X}}_t - \mathbf{X}_t)$ to quantify the final anomaly scores, as formulated below:

$$\mathbf{A}_{\text{map},t} = ((\hat{\mathbf{X}}_t - \mathbf{X}_t) - \boldsymbol{\mu}_m)^T \mathbf{D}^{-1}((\hat{\mathbf{X}}_t - \mathbf{X}_t) - \boldsymbol{\mu}_m) \tag{48}$$

where $\mathbf{D}^{-1}$ denotes the inverse covariance matrix, and $\boldsymbol{\mu}_m$ represents the mean vector of the residuals. The calculated value($\mathbf{A}_{\text{map},t}$) quantifies the degree of anomaly for each pixel.

Algorithm 1 CL-SSMamba

Input: The stream HSIs $\mathbf{X}_1, \ldots, \mathbf{X}_t, \ldots, \mathbf{X}_T$ for T tasks.
Parameter: Window size $w = 3$, channel dimension $C = \min(\mathbf{X}_1, \mathbf{X}_T)$, forgetting rate $\lambda_{AF} = 0.5$, the number of clustering groups $P = 3$, learning rate $lr = 0.00005$.
Continue Training:

```
Initialize the model parameters θ_t.
for t = 1 to T tasks do
    Parse task data: X_t ∈ R^{H×W×C};
    for H training epochs do
        Generate X̄_t vector via data preprocessing module by equation (22), (23) and (24);
        Input X̄_t into SSMamba model to extract spectral-spatial features and obtain the reconstructed background X̂_t;
        if t = 1 then
            Update θ_t by minimizing L_EC and L_recon through equation (45) and (43);
        else
            Update θ_t by minimizing L(θ through equation (47);
        end if
    end for
end for
```

Input: The stream HSIs $\mathbf{X}'_1 \ldots, \mathbf{X}'_t, \ldots, \mathbf{X}'_T$ for T tasks.
Continue Testing:

```
Reconstruct the background X̂'_1, ..., X̂'_T using the trained parameters θ;
Compute anomaly maps through Eq. (48);
```

Output: $\mathbf{A}_{map,t}$.

A.4 Additional Experiments

The experiments below complement the compact results in the main paper. They are organized to answer four questions: whether CL-SSMamba remains stable over the complete task sequence, whether the same behavior holds under temporal shift, which components are responsible for the gains, and what memory/computational cost is introduced by the collaborative learner design.

A.4.1 Full Task-Wise Results on ABU

Table 5 summarizes the results performance of different methods on the ABU dataset in terms of ACC, BWT, FWT, and parameter complexity. CL-SSMamba consistently achieves an ACC of over 96% across all tasks, with the highest value reaching 98.38% in Task 3, demonstrating a substantial performance advantage over CL-AE at 95.12%, CL-TAEF at 64.92%, and CL-GT at 64.25%. In addition, CL-SSMamba maintains a low BWT value throughout all tasks, indicating that the method effectively preserves historical knowledge during the learning of new tasks, thereby significantly alleviates catastrophic forgetting. In terms of parameter analysis, the employment of a dual-learner architecture results in a double increase in parameter volume. Nevertheless, in the continual HAD scenarios, the model demonstrates superior overall performance and achieves an optimal balance among detection accuracy, learning stability, and computational cost.

Table 6 presents the comparative AUC_{BS} results for each algorithm. CL-SSMamba achieves the highest average AUC_{BS} value among the compared methods. Furthermore, it exhibits minimal performance variance as the task sequence increases, indicating that CL-SSMamba effectively suppresses catastrophic forgetting , while maintaining high detection accuracy. Furthermore, it successfully mitigates interference induced by incoming tasks, significantly alleviating the performance degradation on historical tasks.

A.4.2 Retention and Transfer Dynamics on ABU

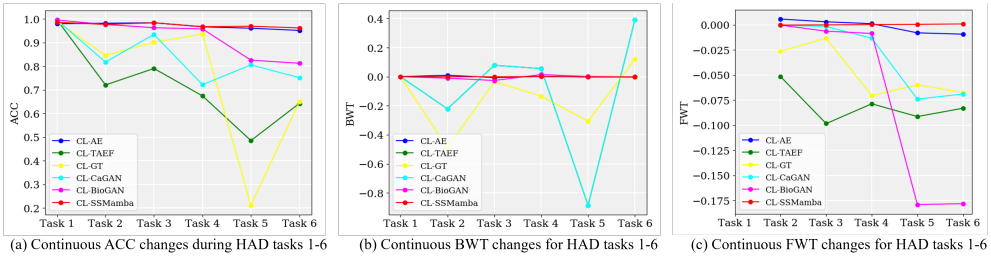
The following analyses visualize how ACC, BWT, FWT, and task-wise detection performance evolve with newly arriving scenes. They complement the compact final-stage table in the main paper by exposing the temporal behavior of catastrophic forgetting.

Table 5: Continual learning for ABU Dataset performance with three evaluation metrics.

ABU Dataset Tasks	Metrics	Method CL-AE	CL-TAEF	CL-GT	CL-CaGAN	CL-BioGAN	CL-SSMamba
2	ACC↑	**0.9817**	0.8459	0.7205	0.8170	0.9765	0.9767
	BWT↑	**0.0097**	-0.2214	-0.4894	-0.3294	-0.0102	-0.0004
	FWT↑	**0.0057**	-0.0516	-0.0264	-0.0005	-0.0004	-0.0004
3	ACC↑	0.9836	0.9011	0.7901	0.9346	0.9627	**0.9838**
	BWT↑	-0.0063	0.0786	-0.0349	0.0864	-0.0262	**-0.0010**
	FWT↑	**0.0029**	-0.0985	-0.0135	-0.0017	-0.0064	-0.0002
4	ACC↑	0.9665	0.9358	0.6742	0.7216	0.9578	**0.9672**
	BWT↑	**0.0016**	0.0555	-0.1337	-0.2644	0.0145	0.0012
	FWT↑	**0.0011**	-0.0788	-0.0707	-0.0135	-0.0087	0.0002
5	ACC↑	0.9606	0.2114	0.4849	0.8055	0.8255	**0.9688**
	BWT↑	-0.0023	-0.8847	-0.3068	**0.1038**	0.0012	-0.0015
	FWT↑	-0.0081	-0.0915	-0.0598	-0.0741	-0.1791	**0.0000**
6	ACC↑	0.9512	0.6492	0.6425	0.7514	0.8123	**0.9620**
	BWT↑	-0.0023	**0.3917**	0.1197	-0.0801	-0.0018	-0.0012
	FWT↑	-0.0094	-0.0832	-0.0675	-0.0690	-0.1783	**0.0008**
-	Training Parameters↓	**77450**	240106	235740	1140426	752208	1179351

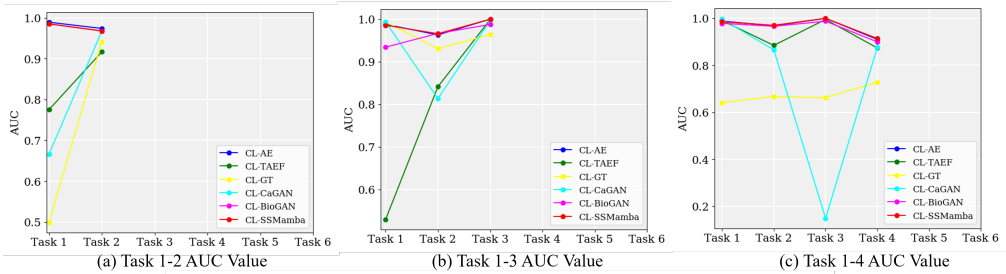
Table 6: AUC-BS Values for Different CL Policies on the ABU Dataset

Method	CL-AE	CL-TAEF	CL-GT	CL-CaGAN	CL-BioGAN	CL-SSMamba
1	0.9174	0.9829	0.9849	**0.9951**	0.9950	0.9622
Avg↑	0.9174	0.9829	0.9849	**0.9951**	0.9950	0.9622
1	0.9656	0.6585	0.1485	0.6460	**0.9835**	0.9635
2	0.8917	0.8467	0.8876	**0.9243**	0.9028	0.8959
Avg↑	0.9287	0.7526	0.5181	0.7852	**0.9432**	0.9297
1	0.9181	0.8873	0.1837	**0.9742**	0.9324	0.9623
2	0.8388	0.8464	0.6921	0.7377	**0.8996**	0.8932
3	0.9053	0.8076	0.9691	**0.9796**	0.9344	0.9513
Avg↑	0.8874	0.8471	0.6150	0.8972	0.9221	**0.9356**
1	0.9280	0.9633	0.4096	**0.9873**	0.9762	0.9619
2	0.8404	0.7679	0.3797	-0.7387	0.8942	**0.8976**
3	0.9067	0.9469	0.4049	-0.5006	0.9304	**0.9515**
4	0.7974	0.8367	0.6900	0.8422	**0.8625**	0.8370
Avg↑	0.8681	0.8787	0.4711	0.1476	**0.9158**	0.9120
1	0.9155	-0.6351	0.6321	0.2512	**0.9828**	0.9611
2	0.8439	0.0809	0.3777	0.6195	**0.8975**	0.8919
3	0.7828	-0.4312	0.3329	0.6203	0.9336	**0.9457**
4	**0.9058**	-0.6588	0.5604	0.4903	0.8638	0.8369
5	0.8697	0.8088	0.0824	0.4450	0.0421	**0.9237**
Avg↑	0.8635	-0.1671	0.3971	0.4853	0.7440	**0.9119**
1	0.9139	0.4122	0.6321	0.9717	**0.9756**	0.9619
2	0.8306	0.6246	0.3777	0.2047	0.8849	**0.8970**
3	0.9120	0.3253	0.3329	0.6392	0.9302	**0.9474**
4	0.7895	0.0218	0.5604	0.7880	**0.8624**	0.8422
5	0.8428	0.4744	0.0824	0.0880	0.0349	**0.9196**
6	0.8076	0.4209	0.8236	0.7117	0.7493	**0.9061**
Avg↑	0.8494	0.3799	0.4682	0.5672	0.7396	**0.9124**



(a) Continuous ACC changes during HAD tasks 1-6 (b) Continuous BWT changes for HAD tasks 1-6 (c) Continuous FWT changes for HAD tasks 1-6

Figure 10: Performance Evaluation of ACC, BWT, and FWT Methods on Continual HAD



(a) Task 1-2 AUC Value (b) Task 1-3 AUC Value (c) Task 1-4 AUC Value

(d) Task 1-5 AUC Value (e) Task 1-6 AUC Value

Figure 11: Performance Evaluation of Catastrophic Forgetting in HAD for Persistent Tasks

Fig.10 illustrates the variation trends of ACC, BWT, and FWT across different methods on the ABU dataset, where the red curve represents the proposed CL-SSMamba. The ACC curve of CL-SSMamba consistently remains above of the compared methods, demonstrating superior performance and high stability. Notably, the BWT value remains close to zero throughout the task sequence, indicating that catastrophic forgetting is effectively alleviated in CL-SSMamba. Furthermore, the FWT becomes positive value after Task 4, which suggests that knowledge acquired from earlier tasks can be effectively transferred to subsequent tasks.

Fig.11 illustrates the fluctuation trends of $\text{AUC}_{(D,F)}$ following the sequential learning of new tasks, where the five subplots correspond to the performance variations of the five comparative methods. It is evident from the observation that CL-SSMamba consistently maintains high anomaly detection accuracy, regardless of the accumulation of learned tasks. This phenomenon proves the robust background modeling capability of CL-SSMamba, enabling it to effectively characterize HSI background distributions and achieve high-quality reconstruction. Furthermore, the results indicate that the model successfully balances the acquisition of new knowledge with the preservation of historical information, thereby alleviating excessive forgetting and sustaining superior detection performance. Moreover, the consistent stability exhibited across sequential tasks further validates the robustness and adaptability of the proposed method in continual learning scenarios.

A.4.3 Qualitative Results on ABU

Fig. 12 presents the qualitative visualization results after the completion of the six-task sequence. CL-SSMamba effectively highlights anomalous regions and achieves high-quality background reconstruction with minimal noise interference, thereby effectively discriminating between anomalies and background. Furthermore, the detected anomalies exhibit sharper and more distinct boundaries

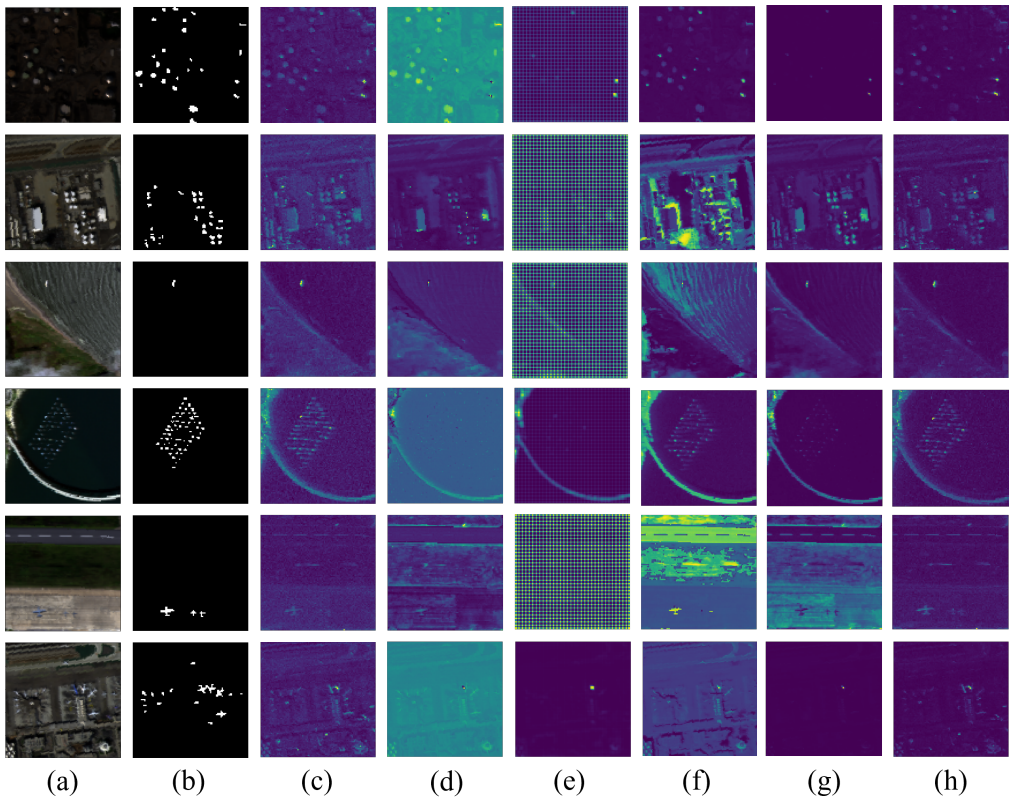


Figure 12: Visualization results of continual HAD over six ABU datasets. From top to bottom: Los Angeles-1, Los Angeles-2, Bay Champagne, San Diego, Gulfport, and Los Angeles-3. (a) ABU pseudo-color image; (b) ground truth annotation; (c) CL-AE; (d) CL-TAEF; (e) CL-GT; (f) CL-CaGAN; (g) CL-BioGAN; (h) CL-SSMamba

compared to those obtained by competing methods. In contrast, CL-GT is compromised by grid-like artifacts, while CL-TAEF and CL-CaGAN suffer from severe background clutter and blurred target boundaries. The experimental results demonstrate that CL-SSMamba not only achieves superior performance in HAD tasks but also yields highly reliable and precise detection results.

A.4.4 Temporal-Shift Results on HAD100

Table 7 presents the quantitative evaluation results on the HAD100 dataset after the completion of continual learning tasks, including ACC, BWT, FWT, and Avg BS from Task 2 to Task 5. The experimental results indicate that CL-TAEF exhibits limited generalization capabilities due to its reliance on specific non-linear parameter settings. While CL-GT demonstrates acceptable performance during the initial tasks, it suffers from significantly exacerbated catastrophic forgetting as the sequential tasks accumulate. Although CL-GT achieves acceptable performance in the early tasks, it suffers from increasingly severe catastrophic forgetting as the task sequence progresses. Furthermore, while CL-CaGAN and CL-BioGAN maintain relatively stable performance, their detection accuracy remains unsatisfactory. In contrast, CL-SSMamba not only achieves high-precision anomaly detection but also demonstrates a lower forgetting rate and stronger forward transfer capability.

A.4.5 Qualitative Results on HAD100

Fig. 13 presents the visualization results of anomaly detection on the HAD100 dataset after the completion of five sequential tasks. It is evident that CL-SSMamba clearly identifies anomalous regions across all tasks while exhibiting robust suppression capabilities against background clutter. In contrast, CL-TAEF and CL-GT exhibit significantly inferior detection performance, characterized

Table 7: Metrics on HAD100 with Different Settings

HAD100	2				3			
Metrics	ACC↑	BWT↑	FWT↑	*Avg BS* ↑	ACC↑	BWT↑	FWT↑	*Avg BS* ↑
CL-TAEF	0.4926	-0.2110	-0.4404	0.2735	0.6861	**0.2657**	-0.4462	0.5179
CL-GT	0.8826	-0.1307	-0.0883	0.7553	0.8756	-0.0684	-0.0418	0.8568
CL-CaGAN	0.9740	-0.0289	-0.0175	**0.9546**	0.9213	-0.0861	-0.0125	0.8940
CL-BioGAN	0.9815	-0.0129	-0.0159	0.9441	0.9833	0.0004	-0.0108	**0.9420**
CL-SSMamba	**0.9919**	**-0.0024**	**0.0017**	0.9063	**0.9904**	-0.0026	**0.0003**	0.8886
HAD100	2				3			
	ACC↑	BWT↑	FWT↑	*Avg BS* ↑	ACC↑	BWT↑	FWT↑	*Avg BS* ↑
CL-TAEF	0.5559	-0.1126	-0.4630	0.3160	0.5618	0.0029	-0.4395	0.2076
CL-GT	0.6406	-0.3542	-0.0283	0.3884	0.8057	**0.1189**	-0.0095	0.5973
CL-CaGAN	0.9418	**0.0014**	-0.0083	0.8821	0.9595	0.0101	0.0059	0.8695
CL-BioGAN	0.9877	0.0003	-0.0073	**0.9305**	**0.9885**	0.0001	**0.0062**	**0.9332**
CL-SSMamba	**0.9926**	-0.0002	**0.0002**	0.8981	0.9827	-0.0012	0.0013	0.8817

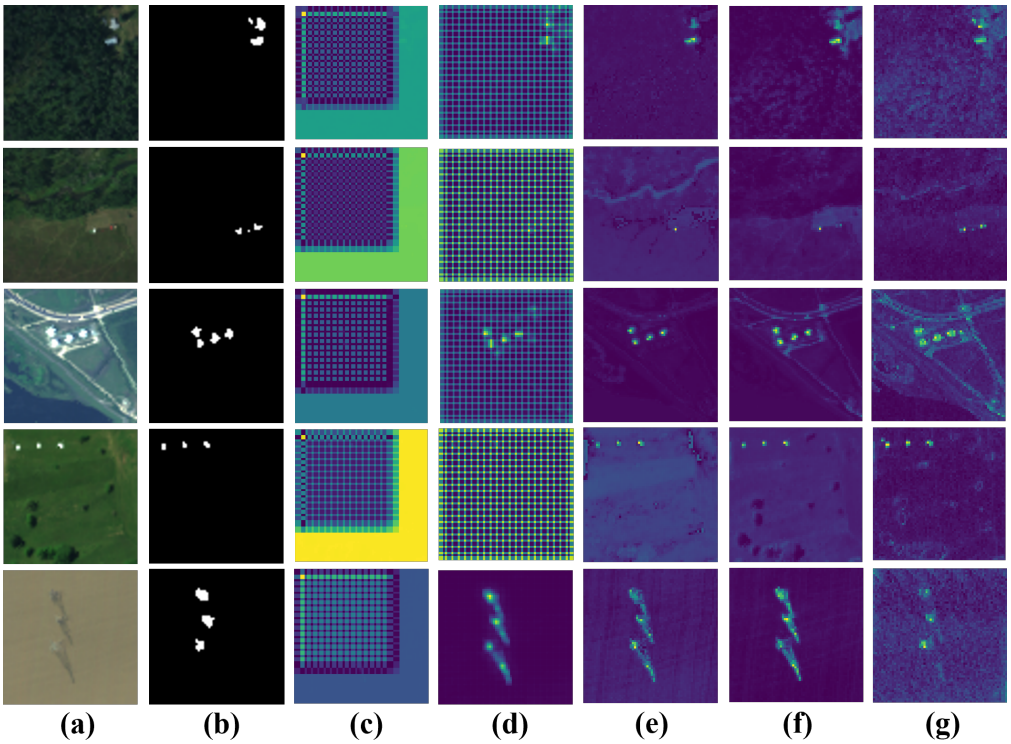


Figure 13: Visualization results of continual HAD over five ABU datasets. From top to bottom: Los Angeles-1, Los Angeles-2, Bay Champagne, San Diego, Gulfport, and Los Angeles-3. (a) ABU pseudo-color image; (b) ground truth annotation; (c) CL-TAEF; (d) CL-GT; (e) CL-CaGAN; (f) CL-BioGAN; (g) CL-SSMamba.

by pronounced grid-like artifacts in their final detection maps. This result indicates that these models fail to effectively preserve spatial continuity and structural information during feature extraction and image reconstruction. Regarding the generative model-based methods, CL-CaGAN and CL-BioGAN, although they demonstrate competitive detection capabilities in continual tasks, varying degrees of deviation persist in their specific results. These deviations manifest as blurred detection boundaries and imprecise anomaly localization in certain scenarios, suggesting inherent bottlenecks in handing cross-task feature transfer that constrain further improvements in accuracy. Overall, these experimental results further validate the effectiveness and stability of CL-SSMamba in OHAD tasks.

A.4.6 IMPACT OF THE NUMBER OF CONTINUAL LEARNERS

Table 8 investigates the impact of the number of continual learners (K) on model performance. Setting $K = 2$ yields an average AUC$_{(D,F)}$ of 97.67% and an AUC$_{BS}$ of 0.9297, representing a substantial improvement over the $K = 1$ configuration (AUC$_{BS}$=0.8379). While increasing K to 3 results in a marginal AUC$_{BS}$ gain (0.9382), it incurs a 50% increase in parameter complexity and exacerbates catastrophic forgetting, as evidenced by the BWT degradation to -0.0134. Consequently, $K = 2$ is adopted as the optimal configuration to strike a balance between detection accuracy and computational efficiency.

Table 8: Values of AUC(D,F) and AUC$_{BS}$ for different K values on the ABU dataset

K	$AUC_{(D,F)}$			K	AUC_{BS}		
	1	2	3		1	2	3
1	**0.9858**	**0.9858**	0.9854	1	0.9627	0.9622	0.9626
ACC	0.9858	0.9858	0.9854	Avg↑	0.9627	0.9622	0.9626
BWT	-	-	-	-	-	-	-
1	0.8248	**0.9854**	0.9721	1	0.7902	**0.9635**	0.9463
2	0.9263	0.9680	0.9669	2	0.8856	**0.8959**	0.8949
ACC	0.8755	0.9767	0.9695	Avg↑	0.8379	0.9297	0.9206
BWT	-0.1610	-0.0004	-0.0134	-	-	-	-
1	0.9083	**0.9857**	0.9856	1	0.8761	**0.9623**	0.9618
2	0.9020	0.9658	0.9673	2	0.7555	0.8932	**0.9029**
3	**0.9998**	**0.9999**	0.9998	3	**0.9108**	**0.9513**	0.9499
ACC	0.9367	0.9838	0.9843	Avg↑	0.8475	0.9356	0.9382
BWT	**0.0296**	-0.0010	0.0070	-	-	-	-
1	0.9842	**0.9852**	0.9846	1	**0.9619**	**0.9619**	0.9591
2	0.9664	0.9699	0.9660	2	0.8942	**0.8976**	0.8934
3	**0.9999**	**0.9998**	**0.9999**	3	**0.9496**	**0.9515**	0.9514
4	0.9134	0.9137	0.9140	4	0.8376	0.8370	0.8454
ACC	0.9659	0.9672	0.9661	Avg↑	0.9108	0.9120	0.9123
BWT	**0.0468**	0.0012	-0.0008	-	-	-	-
1	0.9377	**0.9851**	0.9744	1	0.9169	**0.9611**	0.9513
2	0.9652	0.9648	0.9666	2	0.8787	**0.8919**	0.8853
3	**0.9998**	**0.9999**	**0.9999**	3	**0.9391**	**0.9457**	**0.9509**
4	0.8989	0.9130	0.9085	4	0.8703	0.8369	0.8344
5	**0.9816**	**0.9815**	0.9806	5	**0.9298**	0.9237	0.9261
ACC	0.9567	0.9688	0.9570	Avg↑	0.9070	0.9119	0.9096
BWT	-0.0155	-0.0015	-0.0038	-	-	-	-
1	0.9851	**0.9859**	0.7631	1	0.9615	**0.9619**	0.7161
2	**0.9658**	0.9651	0.8317	2	0.8938	**0.8970**	0.7426
3	**0.9998**	**0.9999**	0.8535	3	**0.9495**	0.9474	0.7035
4	0.9102	0.9108	0.7383	4	0.8372	**0.8422**	0.6856
5	**0.9771**	**0.9764**	0.9680	5	**0.9165**	**0.9196**	0.9022
6	0.9333	0.9338	0.8623	6	**0.9061**	**0.9061**	0.8268
ACC	0.9619	0.9620	0.8361	Avg↑	0.9108	0.9124	0.7628
BWT	-0.1822	-0.0012	-0.1351	-	-	-	-

A.4.7 COMPONENT ABLATION

To validate the individual contributions of specific components within the CL-SSMamba framework, this section conducts a systematic ablation study encompassing six distinct configurations: Spectral Correlation Enhancement (w/o SCE), data preprocessing (w/o SLIC), Local Context Spatial Fusion (w/o LCSF), Spectral Mamba Feature Extraction (w/o LSM), Global Spatial Mamba Feature Extraction (w/o GSM), and the spectral-spatial connection strategy, specifically replacing the proposed parallel design with a serial connection (w/o SSM). Tables 8 and Tables 10 present the quantitative results for $\text{AUC}_{(D,F)}$ and AUC-BS across these configurations. The empirical data reveal that the complete model consistently achieves superior performance across all metrics, demonstrating robust continual detection capabilities. Conversely, the omission of any single module precipitates a significant degradation in performance, thereby underscoring the indispensability of each component. In conclusion, the synergistic integration of all modules optimizes the model's robustness, ensuring an effective balance among catastrophic forgetting mitigation, background suppression and detection accuracy in incremental learning scenarios.

Table 9: $AUC_{(D,F)}$ Values for Different CL Strategies on the ABU Dataset

Method	w/o SCE	w/o SLIC	w/o LCSF	w/o LSM	w/o SSM	w/o GSM	CL-SSMamba
1	0.9856	0.9794	0.9855	0.9849	0.9856	0.9688	**0.9858**
ACC	0.9856	0.9794	0.9855	0.9849	0.9856	0.9688	**0.9858**
BWT	-	-	-	-	-	-	-
1	0.8701	0.9858	0.9816	0.9030	0.9839	0.9691	**0.9854**
2	**0.9687**	0.9672	0.9682	0.9664	0.9670	0.9438	0.9680
ACC	0.9194	0.9765	0.9840	0.9347	0.9755	0.9564	**0.9767**
BWT	-0.1155	**0.0064**	0.0012	-0.0820	-0.0017	0.0004	-0.0004
1	0.9852	0.9849	0.9853	0.9849	0.9856	0.9701	**0.9857**
2	**0.9669**	0.9647	**0.9669**	0.9662	0.9650	0.9367	0.9658
3	0.9998	0.9998	0.9999	0.9999	0.9999	0.9766	**0.9999**
ACC	0.9840	0.9831	**0.9840**	0.9837	0.9835	0.9611	0.9838
BWT	**0.0566**	-0.0017	0.0012	0.0409	-0.0002	-0.0031	-0.0010
1	0.9853	0.9852	0.9527	0.9854	0.9707	0.9712	**0.9852**
2	0.9680	0.9671	0.9674	0.9665	0.9649	0.9347	**0.9699**
3	0.9998	0.9998	0.9998	0.9998	0.9998	0.9724	**0.9998**
4	0.9135	0.9127	0.8901	0.9120	0.8964	0.8397	0.9137
ACC	0.9666	0.9662	0.9525	0.9659	0.9580	0.9295	**0.9672**
BWT	0.0004	0.0009	-0.0107	0.0002	-0.0050	-0.0017	0.0012
1	0.9376	0.9853	0.9851	0.9861	0.9852	0.9510	**0.9851**
2	**0.9675**	0.9666	0.9667	0.9673	0.9662	0.9377	0.9648
3	0.9999	0.9998	0.9999	0.9999	0.9999	0.9785	**0.9999**
4	0.8740	0.9106	0.9112	0.9133	0.9106	0.8242	0.9130
5	0.9793	0.9818	0.9810	0.9813	0.9814	0.9413	**0.9815**
ACC	0.9517	0.9688	0.9687	0.9696	0.9686	0.9265	**0.9688**
BWT	-0.0219	-0.0007	**0.0132**	0.0007	0.0075	-0.0067	-0.0015
1	0.9350	0.9853	0.9215	0.9852	0.9847	0.8745	**0.9859**
2	0.9272	0.9676	0.9659	**0.9679**	0.9661	0.8455	0.9651
3	0.9995	0.9998	0.9999	0.9998	0.9998	0.9477	**0.9999**
4	0.8740	0.9108	0.8631	0.9126	**0.9132**	0.6932	0.9108
5	0.9731	**0.9802**	0.6799	0.9775	**0.9788**	0.4924	0.9764
6	0.9071	0.8610	0.9255	0.9333	0.9316	0.7137	0.9338
ACC	0.9360	0.9508	0.9426	**0.9627**	0.9624	0.7612	0.9620
BWT	-0.0099	**-0.0001**	-0.0227	-0.0009	-0.0001	-0.1559	-0.0012

A.4.8 MEMORY, CAPACITY, AND COMPUTATIONAL EFFICIENCY

The complete CL-SSMamba contains 1,179,351 trainable parameters in the reported two-learner configuration. This is larger than single-learner baselines because the parallel learners provide complementary background representations, but the model does not retain historical raw hyperspectral samples for replay. In the reported RTX 3090 experiments, CL-SSMamba achieves a 15% training-speed improvement relative to CL-CaGAN while maintaining the strongest overall

Table 10: AUC-BS Values for Different CL Strategies on the ABU Dataset

Method	w/o SCE	w/o SLIC	w/o LCSF	w/o LSM	w/o SSM	w/o GSM	CL-SSMamba
1	0.9614	0.9545	0.9632	0.9602	0.9602	0.9536	0.9622
Avg↑	0.9614	0.9545	0.9632	0.9602	0.9602	0.9536	0.9622
1	0.9572	0.9627	0.8358	0.9599	0.9611	0.9560	**0.9635**
2	0.8869	0.8917	0.8970	0.9018	0.8978	**0.9277**	0.8959
Avg↑	0.9221	0.9272	0.8664	0.9309	0.9294	**0.9419**	0.9297
1	0.9610	0.9618	0.9621	0.9605	0.9618	0.9587	**0.9623**
2	0.8939	0.8919	0.9008	0.8940	0.8967	0.9200	0.8932
3	0.9491	0.9481	0.9482	0.9489	0.9450	0.9600	0.9513
Avg↑	0.9347	0.9339	0.9370	0.9345	0.9345	**0.9462**	0.9356
1	0.9221	**0.9627**	0.9624	0.9626	0.9443	0.9567	0.9619
2	0.8973	0.8889	0.9005	0.9009	0.8904	**0.9182**	0.8976
3	0.9464	0.9523	0.9499	0.9434	0.9461	0.9545	0.9515
4	0.8108	0.8350	0.8327	**0.8406**	0.8247	0.8235	0.8370
Avg↑	0.8942	0.9097	0.9114	0.9119	0.9014	**0.9132**	0.9120
1	**0.9630**	0.9621	0.9060	0.9592	0.9618	0.9356	0.9611
2	0.8919	0.8870	0.8956	0.8975	0.8921	**0.9187**	0.8919
3	0.9500	0.9490	0.9503	0.9434	0.9484	**0.9583**	0.9457
4	**0.8400**	0.8397	0.8047	0.8389	0.8393	0.8046	0.8369
5	**0.9298**	0.9297	0.9259	0.9274	0.9279	0.9208	0.9237
Avg↑	**0.9149**	0.9135	0.8965	0.9133	0.9139	0.9076	0.9119
1	0.8901	0.9604	0.9060	**0.9627**	0.9601	0.8451	0.9619
2	**0.9009**	0.8836	0.8144	0.8954	0.8953	0.8115	0.8970
3	0.9480	**0.9493**	0.9298	0.9489	0.9456	0.9132	0.9474
4	0.7921	0.8385	0.8076	0.8374	0.8393	0.6601	**0.8422**
5	**0.9296**	0.9265	0.9147	0.9232	0.9242	0.4364	0.9196
6	0.8978	0.8265	0.8733	0.9040	0.9033	0.6780	**0.9061**
Avg↑	0.8931	0.8975	0.8743	0.9119	0.9113	0.7241	**0.9124**

accuracy-retention trade-off on ABU. Thus, the additional model capacity is exchanged for replay-free continual adaptation rather than for an ever-growing historical sample buffer.

Takeaway. Taken together, the supplementary results support three distinct claims rather than merely repeating the main paper: the multi-learner structure stabilizes sequential adaptation, the spatial-spectral modules are complementary for background reconstruction, and the replay-free design avoids historical raw-data storage while remaining computationally competitive.