Mitigating Asymmetric Boundary Encroachment in Continual Learning of Vision-Language Models

Mingfeng Li[1] **Xinyang Chen**[1✉] **Xiucheng Li**[1] **Weili Guan**[2] **Liqiang Nie**[1]

[1]School of Computer Science and Technology, Harbin Institute of Technology (Shenzhen)
[2]School of Information Science and Technology, Harbin Institute of Technology (Shenzhen)

{limingfeng1900,chenxinyang95,nieliqiang}@gmail.com
{lixiucheng,guanweili}@hit.edu.cn

Abstract

When learning new concepts sequentially, Vision-Language Models (VLMs) face forgetting that degrades both previously acquired tasks and inherent zero-shot capabilities. This challenge is particularly pronounced in the Cross-domain Task-Agnostic Incremental Learning (X-TAIL) setting, where the absence of explicit task identifiers forces diverse concepts to coexist within a unified and crowded multi-modal space. In this context, we identify an underexplored vulnerability termed Asymmetric Boundary Encroachment (ABE). Even when historical feature spaces remain stable, unconstrained new text features actively encroach upon these established sub-spaces during training. To systematically counteract ABE, we propose a unified framework. First, we introduce Analytic Decision Boundary (ADB), which constructs a geometric defense by enforcing an analytic margin derived from cumulative statistics to secure historical multi-modal boundaries. Furthermore, we integrate Orthogonal Visual Adaptation (OVA) and Analytic Ridge Classifier (ARC) to safely evolve the visual backbone and enhance joint inference. Experiments under X-TAIL setting demonstrate that our framework effectively mitigates ABE and consistently achieves new state-of-the-art performance. Our code is available at https://github.com/iLearn-Lab/NeurIPS26-ABE-CL-VLM.

1 Introduction

Vision-Language Models (VLMs), such as CLIP [27], have established a remarkable paradigm for open-vocabulary recognition. However, adapting them to dynamic downstream tasks requires Continual Learning (CL), which inevitably suffers from *catastrophic forgetting* [9, 20], as the model tends to forget prior knowledge upon learning new concepts. In VLMs, the challenge shifts to acquiring new concepts while simultaneously preserving previously learned tasks and the zero-shot transfer ability inherited from pre-training knowledge [44].

This challenge is particularly pronounced under the recently proposed Cross-domain Task-Agnostic Incremental Learning (X-TAIL) setting [39]. X-TAIL requires models to continually learn from diverse datasets and classify images in a unified label space, encompassing both seen and unseen classes without task identifiers. This absence of task isolation closely aligns with real-world scenarios, forcing diverse concepts to coexist and interact within a highly crowded multi-modal joint space.

To preserve previously acquired knowledge while acquiring new concepts, prior works have developed various strategies. Replay and distillation methods [15, 28, 44] rely on historical exemplars or external reference data to preserve the knowledge of historical classes. Parameter-Efficient Fine-Tuning (PEFT) methods [17, 31, 40] isolate new task updates into distinct parameter sub-spaces to mitigate cross-task parameter interference. Alternatively, regression methods [1, 21, 39] utilize recursive ridge

40th Conference on Neural Information Processing Systems (NeurIPS 2026).

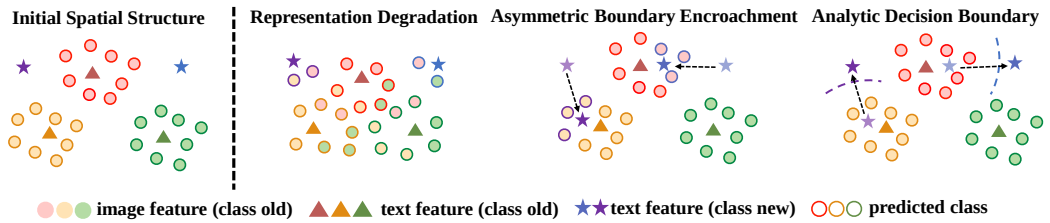


Figure 1: Illustration of Asymmetric Boundary Encroachment and our proposed solution. From left to right: (1) Initial Spatial Structure shows the well-separated feature space of historical classes. (2) Representation Degradation shows the traditional degradation of historical features. (3) Asymmetric Boundary Encroachment shows new text features encroaching upon established historical spaces. (4) Analytic Decision Boundary blocks this encroachment with a rigid geometric margin.

regression to construct classifiers, yet they typically freeze the feature extractor, limiting feature adaptation for new concepts in order to preserve historical feature representations.

However, under the X-TAIL setting, we observe that acquiring new concepts can pose an additional risk to preserving prior knowledge. Existing methods mainly focus on maintaining stable historical multi-modal alignments and feature representations to preserve prior knowledge. Yet within a unified and crowded joint feature space, even when the feature spaces of historical classes remain stable, newly optimized text features can still encroach upon these established sub-spaces during training. We identify this underlying vulnerability as **Asymmetric Boundary Encroachment**.

This asymmetric encroachment exhibits three distinctive properties: **first**, it persists even when the representations of historical classes remain stable; **second**, it is an optimization-driven dynamic process, absent before training but rapidly emerging during optimization; **third**, it occurs across a broad spectrum of classes, rather than being limited to highly similar pairs. Detailed empirical validation and visualizations of these properties are provided in Appendix A, B and C. This discovery extends the conventional view of preventing forgetting in VLMs: effective knowledge retention requires not only preserving the representations of historical classes, but also defending their spatial territory against the expansion of new tasks.

To address Asymmetric Boundary Encroachment while continually acquiring new concepts, we propose a unified framework. First, we introduce the Analytic Decision Boundary (ADB), which constructs a geometric defense by enforcing a margin derived from cumulative statistics to secure the multi-modal decision boundaries of historical classes. Building on this protection, we enhance new concept acquisition from the visual side. Specifically, we integrate Orthogonal Visual Adaptation (OVA) to safely evolve the visual backbone while preserving historical visual representations, and pair it with Analytic Ridge Classifier (ARC), which leverages fine-grained visual features to complement the VLM's vision-language similarity logits. Unified by shared cumulative feature statistics, these components secure historical decision boundaries while improving joint inference performance.

In summary, our main contributions are as follows:

- We identify Asymmetric Boundary Encroachment (ABE), an optimization-driven vulnerability in Vision-Language continual learning, where newly optimized text features encroach upon established historical sub-spaces even when historical feature spaces remain stable.
- We propose Analytic Decision Boundary (ADB) to mitigate Asymmetric Boundary Encroachment. Crucially, ADB shares cumulative sufficient statistics with our auxiliary modules, enabling a unified framework that simultaneously secures decision boundaries and enhances downstream adaptability.
- Experiments under the X-TAIL setting demonstrate that our method effectively mitigates Asymmetric Boundary Encroachment and achieves new state-of-the-art performance.

2 Related Work

Continual Learning in Vision-Language Models. Vision-Language Models (VLMs) [27] capture rich multi-modal representations, facilitating downstream adaptation in Continual Learning (CL).

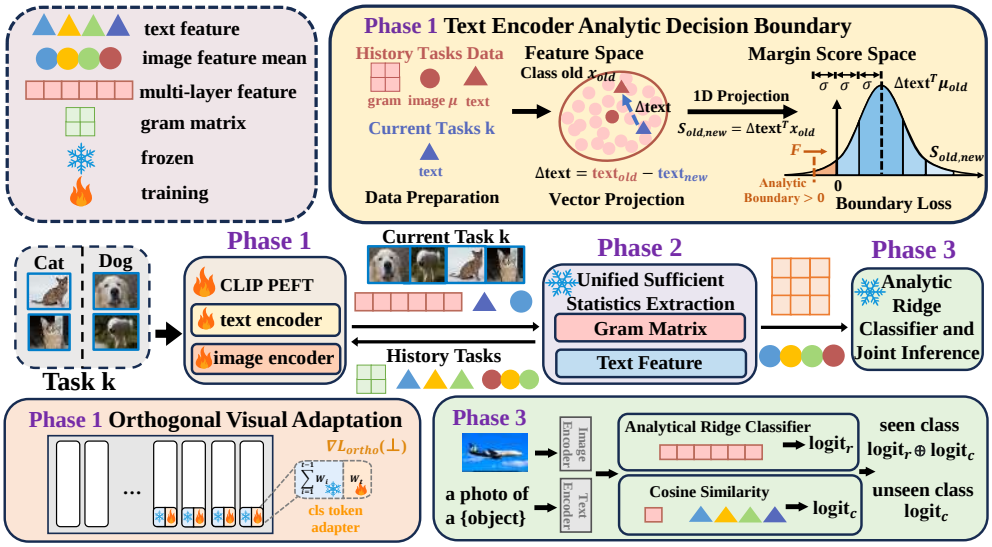


Figure 2: Method Overview. Phase 1 employs Analytic Decision Boundary to protect historical decision boundaries with an analytic margin against new text features (yellow), while Orthogonal Visual Adaptation adapts the visual encoder under orthogonality constraints (orange); Phase 2 extracts shared Gram/text statistics; Phase 3 performs Analytic Ridge Classifier and joint inference (green).

Traditional CL evaluates models in isolated domains, using paradigms like Class-Incremental Learning (CIL) [28] or Task-Incremental Learning (TIL) [12]. Recently, VLM transferability has shifted evaluations toward joint multi-domain scenarios. For example, Multi-domain Task-Incremental Learning (MTIL) [44] learns sequentially across domains but still requires inference task identities. To provide a more realistic evaluation, Cross-domain Task-Agnostic Incremental Learning (X-TAIL) [39] removes task hints, requiring image classification over a joint label space of seen and unseen classes. In this work, we specifically adopt the X-TAIL setting.

Continual Learning Methods. Traditionally, continual learning (CL) approaches are broadly categorized into replay-based, regularization-based, and architecture-based methods [5, 33]. Replay-based methods [3, 16, 28, 30] buffer or synthesize past data but often violate privacy and memory constraints. Regularization-based methods [13, 15, 41] constrain parameter updates via distillation [15] or weight consolidation [13, 41], yet struggle with distribution shifts across different domains. Architecture-based methods [10, 19, 35] dynamically allocate task-specific parameters to prevent interference. For Vision-Language Models (VLMs), parameter-efficient fine-tuning is widely adopted, primarily through prompt-based techniques [23, 31, 32, 35, 37] or adapter-based methods [10, 34, 40, 42]. Diverging from gradient-based tuning, regression methods [21, 24] provide an exemplar-free alternative by applying Ridge Regression with second-order feature statistics. Following this, LayUP [1] leverages multi-layer representations, and RAIL [39] adapts the regression framework for the X-TAIL protocol, but these regression approaches typically sacrifice visual plasticity by freezing the image encoder. Concurrently, projection-based methods [26, 29, 43] use similar statistics to orthogonally regularize weight updates. However, under exemplar-free conditions, these approaches struggle to prevent Asymmetric Boundary Encroachment in the joint embedding space during training optimization, as unconstrained new text features actively encroach upon historical multi-modal spaces. To address this, our framework introduces the Analytic Decision Boundary (ADB) to mitigate this vulnerability, sharing cumulative statistics with auxiliary modules to simultaneously secure decision boundaries and enhance downstream adaptability.

3 Method

3.1 Problem Setting and Overview

Problem Setting. Following the Cross-domain Task-Agnostic Incremental Learning (X-TAIL) protocol [39], a vision-language model learns sequentially from K tasks $\{T_1, T_2, \ldots, T_K\}$. Each task

$T_k = (\mathcal{D}_k, \mathcal{C}_k)$ consists of training images $\mathcal{D}_k$ and corresponding class names $\mathcal{C}_k$. During inference, the model must classify images from seen or unseen domains without task-identity hints, evaluating on the joint label space $\mathcal{C}_{test} = \bigcup_{i=1}^{k} \mathcal{C}_i \cup \mathcal{C}_U$, where $\mathcal{C}_U$ denotes unseen classes.

Overview. As illustrated in Figure 2, our three-phase framework processes each task through training, statistic extraction, and joint inference. In **Phase 1**, given the current task T_k, we adapt the CLIP text and visual encoders using the historical statistics cached in Phase 2 of previous tasks: Analytic Decision Boundary (ADB, Section 3.2) protects historical decision boundaries by enforcing an analytic margin against newly optimized text features, while Orthogonal Visual Adaptation (OVA, Section 3.3) safely evolves the visual encoder. In **Phase 2**, after training, we perform unified sufficient statistics extraction: text features, class visual feature means, and the Gram matrices are collected in a single forward pass. In **Phase 3**, during inference, these accumulated statistics support Analytic Ridge Classifier (ARC, Section 3.3) and adaptive joint inference (Section 3.3), where visual analytic logits complement the VLM similarity logits over the unified label space.

All modules share the class means $\boldsymbol{\mu}_c$ and the hierarchical Gram matrices extracted in Phase 2. For N unnormalized visual features $\mathbf{X} \in \mathbb{R}^{N \times D}$, the core uncentered Gram matrix is defined as $\mathbf{G} = \mathbf{X}^\top \mathbf{X} \in \mathbb{R}^{D \times D}$. To avoid notation clutter, we instantiate this shared statistic into three proxies according to data scope and feature hierarchy, as summarized in Table 1.

Table 1: Summary of Hierarchical Gram Matrices.

Proxy	Scope (Subscript)	Hierarchy (Superscript)	Primary Function	Section
$\mathbf{G}_c^{(L)}$	Class c	Final layer L	Analytically Bounding the Semantic Space	3.2
$\mathbf{G}_k^{(l)}$	Up to task k	Specific layer l	Orthogonal Visual Adaptation	3.3
$\mathbf{G}_k^{(\mathcal{S})}$	Up to task k	All selected layers $\mathcal{S}$	Joint Analytic Inference	3.3

3.2 Analytic Decision Boundary via Low-Rank Approximation

Probabilistic Modeling of the Margin. To preserve the learned representations of historical classes, we cache their optimal text features as frozen anchors $\mathbf{w}_c$. However, the multi-modal joint space remains vulnerable to Asymmetric Boundary Encroachment (ABE). To counteract this, we enforce an analytic geometric boundary using cumulative feature statistics.

Specifically, for each task k, we fine-tune the text encoder and cache the resulting anchors $\mathbf{w}_c$. To formulate the geometric boundary, we model the unnormalized visual features of a past category $c \in \mathcal{C}_{\text{old}}$ as a Gaussian random vector $\mathbf{x}_c \sim \mathcal{N}(\boldsymbol{\mu}_c, \boldsymbol{\Sigma}_c)$. For a sample $\mathbf{x}_c$ to be correctly classified against an incoming category $c' \in \mathcal{C}_{\text{new}}$, its projection onto the ground-truth $\mathbf{w}_c$ must exceed its projection onto the newly learned text weight $\mathbf{w}_{c'}$. During derivation, all textual weights ($\mathbf{w}_c$ and $\mathbf{w}_{c'}$) are strictly ℓ_2-normalized to align with cosine-similarity-based inference. Defining the pairwise weight differential as $\Delta\mathbf{w}_{c,c'} = \mathbf{w}_c - \mathbf{w}_{c'}$, the relative classification margin is formulated as:

$$S_{c,c'} = \Delta\mathbf{w}_{c,c'}^\top \mathbf{x}_c. \tag{1}$$

Since $S_{c,c'}$ is a linear combination of jointly Gaussian variables, it remains a scalar Gaussian random variable with the analytical expectation and variance:

$$\mu_{S_{c,c'}} = \Delta\mathbf{w}_{c,c'}^\top \boldsymbol{\mu}_c, \quad \sigma^2_{S_{c,c'}} = \Delta\mathbf{w}_{c,c'}^\top \boldsymbol{\Sigma}_c \Delta\mathbf{w}_{c,c'}. \tag{2}$$

To bound the margin distribution and mitigate Asymmetric Boundary Encroachment, we establish a probabilistic lower bound parameterized by a margin multiplier γ:

$$\mu_{S_{c,c'}} - \gamma\sigma_{S_{c,c'}} > 0, \tag{3}$$

where γ dictates the strictness of the boundary. In our experiments, we set $\gamma = 3$, which covers approximately 99.7% of the expected variations. The sensitivity to γ is analyzed in Fig. 7.

Low-Rank Approximation for Boundary Enforcement.

Computing the margin variance $\sigma^2_{S_{c,c'}}$ via the dense covariance $\boldsymbol{\Sigma}_c \in \mathbb{R}^{D \times D}$ incurs a prohibitive $\mathcal{O}(|\mathcal{C}|D^2)$ memory overhead. To address this issue, we employ a low-rank approximation method to compress the boundary constraints.

We construct the sample covariance using the final-layer foundational statistic $\mathbf{G}_c^{(L)} = (\mathbf{X}_c^{(L)})^\top \mathbf{X}_c^{(L)}$ and the mean of its final-layer features $\boldsymbol{\mu}_c$:

$$\boldsymbol{\Sigma}_c = \frac{1}{N_c - 1} \left(\mathbf{G}_c^{(L)} - N_c \boldsymbol{\mu}_c (\boldsymbol{\mu}_c)^\top \right), \tag{4}$$

where N_c is the class sample count. To adaptively truncate the feature space, we apply eigendecomposition and dynamically determine the truncation dimension M by retaining the top eigen-components that satisfy a predefined energy retention threshold η. The choice of η is discussed in Fig. 3(b). This yields the approximation $\boldsymbol{\Sigma}_c \approx \mathbf{U}_{c,M} \boldsymbol{\Lambda}_{c,M} \mathbf{U}_{c,M}^\top$, defining a lightweight generator matrix:

$$\mathbf{P}_c = \mathbf{U}_{c,M} \boldsymbol{\Lambda}_{c,M}^{1/2}. \tag{5}$$

By substituting $\boldsymbol{\Sigma}_c \approx \mathbf{P}_c \mathbf{P}_c^\top$, the margin's variance calculation simplifies from a heavy quadratic form to a computationally efficient ℓ_2-norm:

$$\sigma_{S_{c,c'}}^2 \approx \Delta \mathbf{w}_{c,c'}^\top (\mathbf{P}_c \mathbf{P}_c^\top) \Delta \mathbf{w}_{c,c'} = \left\| \mathbf{P}_c^\top \Delta \mathbf{w}_{c,c'} \right\|_2^2. \tag{6}$$

Building upon this efficient approximation and the established margin lower bound, we penalize old-new class pairs whose estimated margin lower bound is non-positive to mitigate the misclassification. We formalize this constraint into a deterministic hinge loss:

$$\mathcal{L}_{\text{boundary}} = \sum_{c \in \mathcal{C}_{\text{old}}} \sum_{c' \in \mathcal{C}_{\text{new}}} \max \left(0,\ \gamma \left\| \mathbf{P}_c^\top \Delta \mathbf{w}_{c,c'} \right\|_2 - \Delta \mathbf{w}_{c,c'}^\top \boldsymbol{\mu}_c \right). \tag{7}$$

This formulation establishes an analytic geometric constraint that effectively mitigates Asymmetric Boundary Encroachment. Furthermore, the 3σ threshold inherently directs the loss toward informative boundary variations while robustly filtering out low-probability tail samples. The complexity and scalability of ADB are discussed in Section G.

3.3 Orthogonal Visual Adaptation and Joint Inference

While ADB mitigates Asymmetric Boundary Encroachment, we further unleash CLIP's visual potential through a synergistic strategy. To safely evolve visual backbone, OVA reuses the cumulative Gram statistics required by ARC to enforce orthogonal weight updates. ARC can thus exploit adapted visual features as a pure-vision complement to VLM similarity logits for improved joint inference, unlike traditional ridge-regression methods that commonly freeze backbones for feature stability.

Orthogonal Visual Adaptation (OVA). To safely adapt to novel tasks without compromising the stable historical visual feature space, we introduce OVA. Specifically, we insert incremental linear adapters after the `[CLS]` token within a subset of the deepest visual layers, denoted as $\mathcal{S}_{\text{adapt}}$, which is set to the last six visual layers in our implementation. For an incoming task k, let $\{\mathbf{W}_l^{(k)}\}_{l \in \mathcal{S}_{\text{adapt}}}$ denote the set of newly initialized adapter weights. We freeze the accumulated historical weights at each layer, defined as $\mathbf{W}_{\text{old},l} = \sum_{i=1}^{k-1} \mathbf{W}_l^{(i)}$, and update only the new layer-wise adapters $\mathbf{W}_l^{(k)}$.

To mitigate feature drift, updates to $\mathbf{W}_l^{(k)}$ are constrained to be orthogonal to the principal feature directions of previous tasks. For each adapted layer $l \in \mathcal{S}_{\text{adapt}}$, the Gram matrix for each task is independently computed from its corresponding classes: $\mathbf{G}_i^{(l)} = \sum_{c \in \mathcal{C}_i} \mathbf{G}_c^{(l)}$, where $\mathbf{G}_c^{(l)} = (\mathbf{X}_c^{(l)})^\top \mathbf{X}_c^{(l)} \in \mathbb{R}^{D_l \times D_l}$ is derived from the layer-l features $\mathbf{X}_c^{(l)}$ of all N_c samples in class c. We apply eigendecomposition to each task's independent Gram matrix: $\mathbf{G}_i^{(l)} = \mathbf{U}_{i,l} \boldsymbol{\Lambda}_{i,l} \mathbf{U}_{i,l}^\top$. By retaining the top-$M$ eigenvectors $\mathbf{U}_{i,l,M}$ and their corresponding eigenvalues $\boldsymbol{\Lambda}_{i,l,M}$ based on an energy retention threshold, we effectively distill the principal feature directions of past tasks. During the adaptation for incoming task k, we project the new weights against the feature directions of all previous tasks $i \in \{1, \ldots, k-1\}$, defining the projection loss as:

$$\mathcal{L}_{\text{ortho}} = \sum_{l \in \mathcal{S}_{\text{adapt}}} \sum_{i=1}^{k-1} \left\| \mathbf{W}_l^{(k)} \mathbf{U}_{i,l,M} \boldsymbol{\Lambda}_{i,l,M}^{1/2} \right\|_F^2. \tag{8}$$

OVA introduces a spectrum-weighted soft regularization that suppresses the responses of incrementally introduced visual adapters along historical feature directions. By constraining adapter responses

without fixing their update subspaces, OVA mitigates historical feature drift while retaining flexibility for incremental visual adaptation.

Analytic Ridge Classifier (ARC). Building upon the highly preserved visual space maintained by OVA, we further introduce ARC to enhance joint inference performance. By exploiting fine-grained vision features, ARC acts as a pure-vision analytic counterpart to complement the VLM's vision-language similarity logits.

Let $\mathbf{\Phi} = \tilde{\phi}_{\mathcal{S}}(\mathbf{X}) \in \mathbb{R}^{N \times LD}$ denote the multi-layer visual features for N samples, where $\mathcal{S}$ contains L selected layers (the last six in our implementation; the choice is discussed in Fig. 3(c)) and each layer produces D-dimensional features. Let $\mathbf{Y} \in \{0,1\}^{N \times |\mathcal{C}|}$ denote the corresponding one-hot label. The classification objective is:

$$\min_{\mathbf{W}} \|\mathbf{Y} - \mathbf{\Phi}\mathbf{W}\|_F^2 + \lambda \|\mathbf{W}\|_F^2, \tag{9}$$

where $\mathbf{W} \in \mathbb{R}^{LD \times |\mathcal{C}|}$ is the weight matrix and λ is a regularization parameter.

The standard solution is analytically given by $\mathbf{W}^* = (\mathbf{\Phi}^\top \mathbf{\Phi} + \lambda \mathbf{I})^{-1} \mathbf{\Phi}^\top \mathbf{Y}$. We recursively accumulate the global multi-layer Gram matrix: $\mathbf{G}_k^{(\mathcal{S})} = \mathbf{G}_{k-1}^{(\mathcal{S})} + \sum_{c \in \mathcal{C}_k} \mathbf{G}_c^{(\mathcal{S})}$, where $\mathbf{G}_c^{(\mathcal{S})} = \tilde{\phi}_{\mathcal{S}}(\mathbf{X}_c)^\top \tilde{\phi}_{\mathcal{S}}(\mathbf{X}_c) \in \mathbb{R}^{LD \times LD}$ is the full uncentered Gram matrix for class c using concatenated multi-layer features. Concurrently, we accumulate the label statistics $\mathbf{S}^{(k)} = \left[\mathbf{S}^{(k-1)}, \left[N_c \boldsymbol{\mu}_c^{(\mathcal{S})}\right]_{c \in \mathcal{C}_k}\right]$, where $\boldsymbol{\mu}_c^{(\mathcal{S})}$ denotes the mean of the multi-layer features for class c. These serve as algebraic proxies, where $\mathbf{G}_k^{(\mathcal{S})} = \mathbf{\Phi}^\top \mathbf{\Phi}$ and $\mathbf{S}^{(k)} = \mathbf{\Phi}^\top \mathbf{Y}$. The analytic logits for a test sample $\mathbf{x}$ are derived as:

$$\mathbf{z}^{\text{ridge}} = \tilde{\phi}_{\mathcal{S}}(\mathbf{x})^\top \left(\mathbf{G}_k^{(\mathcal{S})} + \lambda \mathbf{I}\right)^{-1} \mathbf{S}^{(k)}. \tag{10}$$

Adaptive Joint Inference. To effectively fuse the predictions of the VLM and the Analytic Ridge Classifier, we propose a distribution-aware fusion. We use the average of the frozen and adapted encoder outputs as the visual feature, and compute cosine similarities over the joint label space $\mathcal{C}_{seen} \cup \mathcal{C}_U$, where $\mathcal{C}_{seen}$ and $\mathcal{C}_U$ denote the seen and unseen class sets, respectively. Seen-class text embeddings are averaged from the original and fine-tuned vectors, while unseen-class embeddings remain original. Routing is determined by the top-1 VLM prediction: if the highest-similarity class belongs to $\mathcal{C}_U$, we directly output the fused-feature VLM prediction; otherwise, the sample is assigned to $\mathcal{C}_{seen}$, where the Ridge Classifier is weighted by its predictive uncertainty from the regularized Gram matrix. To eliminate drift from varying sample sizes, we normalize this uncertainty using the intrinsic energy scale $\tau = \frac{1}{D_{feat}} \text{Tr}(\mathbf{G}_{\text{reg}})$:

$$u_{\text{scaled}}(\mathbf{x}) = \tau \cdot \left(\tilde{\phi}_{\mathcal{S}}(\mathbf{x})^\top \mathbf{G}_{\text{reg}}^{-1} \tilde{\phi}_{\mathcal{S}}(\mathbf{x})\right). \tag{11}$$

where $\tilde{\phi}_{\mathcal{S}}(\mathbf{x})$ is the test feature, D_{feat} is the feature dimension, and $\mathbf{G}_{\text{reg}} = \mathbf{G}_k^{(\mathcal{S})} + \lambda \mathbf{I}$.

This scaled uncertainty is mapped into a confidence score $w_{\text{conf}}(\mathbf{x}) = \exp\left(-u_{\text{scaled}}(\mathbf{x})/2\right) \in (0, 1]$. The final prediction is a combination of the VLM cosine similarities $\mathbf{z}^{\text{clip}}$ and the pure-vision analytic scores $\mathbf{z}^{\text{ridge}}$:

$$\mathbf{z}^{\text{final}} = (1 - w_{\text{conf}}(\mathbf{x})) \cdot \mathbf{z}^{\text{clip}} + w_{\text{conf}}(\mathbf{x}) \cdot \mathbf{z}^{\text{ridge}}. \tag{12}$$

By integrating ridge classification on incrementally adapted multi-layer visual features with sample-dependent uncertainty-aware fusion, ARC provides a complementary, text-independent visual prediction signal to mitigate interference from newly optimized text anchors.

4 Experiments

4.1 Experimental Setting

Benchmark. Following the Cross-domain Task-Agnostic Incremental Learning (X-TAIL) protocol [39], we evaluate our proposed framework on a sequence of 10 image classification datasets from diverse domains: Aircraft [18], Caltech101 [8], DTD [4], EuroSAT [11], Flowers [22], Food [2],

Table 2: Comparison of different CL methods on X-TAIL 16-shot for each task in terms of "Transfer", "Average", and "Last" scores (%). Results for our method and LADA are averaged over 5 random seeds. The best results are highlighted in **bold**.

	Method	Aircraft	Caltech101	DTD	EuroSAT	Flowers	Food101	MNIST	Pets	Cars	Sun397	Average
	Zero-shot	23.8	74.3	36.4	37.4	64.1	83.4	43.9	87.8	65.5	60.8	57.7
Transfer	LwF [15]	–	66.6	26.9	19.5	51.0	78.4	26.6	68.9	35.5	56.1	47.7
	WiSE-FT [36]	–	70.1	31.9	25.3	56.3	79.8	29.9	74.9	45.6	56.8	52.3
	ZSCL [44]	–	73.3	32.6	36.8	62.1	83.8	42.1	83.6	56.5	60.2	59.0
	MoE-Adapters [40]	–	71.0	34.9	19.2	63.0	**86.6**	20.0	87.2	63.7	58.6	56.0
	LADA [17]	–	75.0	36.2	36.2	**66.5**	84.1	40.1	88.0	65.4	61.6	61.5
	Ours	–	**76.6**	**36.7**	**42.2**	65.3	84.1	**49.1**	**88.4**	**65.5**	**62.2**	**63.4**
Average	LwF [15]	24.7	79.7	38.3	36.9	63.9	81.0	36.5	71.9	42.7	56.7	53.2
	WiSE-FT [36]	27.1	76.5	40.9	31.3	68.7	81.6	31.4	74.7	51.7	58.4	54.2
	ZSCL [44]	36.0	75.0	40.7	40.5	71.0	85.3	46.3	83.3	60.7	61.5	60.0
	MoE-Adapters [40]	43.6	77.9	52.1	34.7	75.9	**86.3**	45.2	87.4	66.6	60.2	63.0
	Primal-RAIL [39]	44.6	89.5	56.1	69.0	83.8	85.0	63.2	88.9	68.6	62.2	71.1
	Dual-RAIL [39]	45.4	89.4	56.4	69.6	84.0	85.0	63.5	88.8	68.8	62.3	71.3
	LADA [17]	48.8	89.2	60.7	72.0	84.6	85.2	61.1	89.6	69.3	63.0	72.4
	Ours	**53.2**	**92.0**	**61.3**	**75.6**	**85.2**	85.6	**67.7**	**90.0**	**69.5**	**63.8**	**74.4**
Last	LwF [15]	20.9	83.1	47.5	38.2	75.5	84.7	50.1	78.0	75.8	74.6	62.8
	WiSE-FT [36]	21.8	76.8	42.9	20.8	77.5	84.9	30.7	76.6	75.8	72.5	58.0
	ZSCL [44]	33.1	75.3	43.5	35.2	74.6	**87.4**	50.4	84.2	77.3	73.4	63.4
	MoE-Adapters [40]	43.2	78.7	57.6	32.8	79.4	86.0	86.7	87.8	78.2	74.2	70.5
	Primal-RAIL [39]	44.2	94.6	66.8	85.9	96.3	86.8	91.6	91.5	80.6	75.4	81.4
	Dual-RAIL [39]	45.5	94.6	68.6	87.7	97.2	86.9	92.8	91.4	81.9	75.9	82.3
	LADA [17]	49.4	93.2	69.1	87.3	97.0	87.1	92.6	**93.6**	84.8	76.3	83.0
	Ours	**52.9**	**94.7**	**69.3**	**89.9**	**98.4**	87.2	**95.6**	**93.6**	**85.5**	**77.5**	**84.4**

MNIST [6], OxfordPet [25], StanfordCars [14], and SUN397 [38]. This configuration yields a comprehensive and unified label space encompassing over a thousand distinct classes. Under the X-TAIL setting, the model sequentially learns from these domains. Crucially, during inference at any given step, the model is required to classify test images across the joint label space of both seen and unseen domains ($\mathcal{C}_{test} = \mathcal{C}_{seen} \cup \mathcal{C}_{unseen}$) without relying on any task-identity hints.

Evaluation Metrics. To assess the model's performance, we employ three standard metrics in the X-TAIL framework [39]: (1) **Transfer**, which measures the preserved zero-shot generalization on unseen domains prior to their incremental learning; (2) **Average**, which assesses overall stability by averaging the accuracy across all domains at every incremental step; and (3) **Last**, which evaluates the final knowledge retention and adaptability across all domains after completing the entire training sequence. Detailed mathematical formulations for these metrics and implementation details are provided in Appendices D and E.

4.2 Main Results

In this section, we compare our proposed method against state-of-the-art continual learning approaches under the X-TAIL setting. As summarized in Tables 2 and 3, our approach demonstrates consistent superiority across 16-shot and full-shot scenarios. Results for ours and LADA are averaged over 5 random seeds (see Appendix J for standard deviations). Specifically, under the 16-shot setting, our method achieves 63.4% on the **Transfer**, 74.4% on the **Average**, and 84.4% on the **Last** metric, outperforming the previous best approach by 1.9%, 2.0%, and 1.4%, respectively. Scaling to the full-shot regime, our framework maintains this lead by securing the highest overall scores (63.1%, 76.5%, and 87.5%) across the same metrics. These improvements demonstrate that our framework effectively stabilizes multi-modal classification boundaries across the joint label space, optimizing the trade-off between integrating new domain knowledge and preserving previously acquired knowledge. We omit RAIL's Transfer results because its reported protocol uses vanilla CLIP zero-shot accuracy, and we exclude Dual-RAIL from the full-shot setting because storing all features incurs prohibitive computational and GPU memory overhead. Additional results for an alternative task sequence (Order-II) and computational efficiency evaluations are provided in Appendices I and F.

Table 3: Comparison of different CL methods on X-TAIL full-shot for each task in terms of "Transfer", "Average", and "Last" scores (%). Results for our method and LADA are averaged over 5 random seeds. The best results are highlighted in **bold**.

	Method	Aircraft	Caltech101	DTD	EuroSAT	Flowers	Food101	MNIST	Pets	Cars	Sun397	Average
	Zero-shot	23.8	74.3	36.4	37.4	64.1	83.4	43.9	87.8	65.5	60.8	57.7
Transfer	LwF [15]	–	62.4	27.8	10.7	52.0	76.0	25.4	68.3	30.4	54.5	45.3
	WiSE-FT [36]	–	59.9	25.4	10.2	43.9	67.9	29.4	57.4	24.1	50.3	40.9
	ZSCL [44]	–	71.3	33.8	33.0	**66.2**	85.2	40.2	81.9	57.3	**62.5**	59.0
	MoE-Adapters [40]	–	69.5	30.8	19.0	60.3	**85.6**	43.7	85.6	55.5	57.3	56.4
	LADA [17]	–	75.3	35.9	36.7	65.9	84.0	44.6	88.0	65.4	61.2	61.9
	Ours	–	**76.6**	**36.3**	**42.3**	64.6	83.8	**48.1**	**88.4**	**65.7**	62.0	**63.1**
Average	LwF [15]	25.8	81.2	48.1	33.1	57.0	75.1	54.9	74.5	35.5	57.1	54.2
	WiSE-FT [36]	14.9	79.8	45.3	17.1	55.7	70.1	52.0	68.0	35.6	53.3	49.2
	ZSCL [44]	39.7	80.8	52.9	40.8	79.3	**88.0**	51.4	85.5	62.9	**64.1**	64.5
	MoE-Adapters [40]	52.4	79.4	57.7	42.7	81.1	86.6	64.8	86.7	61.3	59.0	67.2
	Primal-RAIL [39]	48.8	89.6	59.0	74.4	84.0	86.1	65.6	89.5	68.9	62.3	72.8
	LADA [17]	53.1	**93.7**	67.0	77.9	**85.4**	86.7	64.8	89.9	69.8	62.8	75.1
	Ours	**59.7**	93.2	**68.3**	**79.9**	**85.4**	86.5	**68.2**	**90.3**	**70.2**	63.6	**76.5**
Last	LwF [15]	9.6	77.1	55.3	38.7	60.5	83.1	**99.5**	85.9	49.6	80.0	63.9
	WiSE-FT [36]	18.1	84.9	53.4	27.0	69.6	88.0	88.4	91.5	76.7	**80.2**	67.8
	ZSCL [44]	33.8	80.4	60.2	31.1	85.8	**91.3**	80.4	93.7	84.9	79.0	72.1
	MoE-Adapters [40]	51.9	79.0	64.2	51.5	95.1	87.6	96.4	89.1	84.4	74.0	77.3
	Primal-RAIL [39]	45.8	94.1	70.7	94.2	96.5	89.0	98.1	93.5	82.0	76.5	84.0
	LADA [17]	55.3	**96.2**	76.1	**95.7**	98.4	89.6	98.8	94.6	87.5	77.3	86.9
	Ours	**58.9**	95.7	**77.1**	95.6	**99.2**	89.0	98.3	**94.7**	**88.0**	78.6	**87.5**

Table 4: Ablation study of proposed components on 16-shot and full-shot X-TAIL benchmarks. OVA: Orthogonal Visual Adaptation; ADB: Analytic Decision Boundary; ARC: Analytic Ridge Classifier.

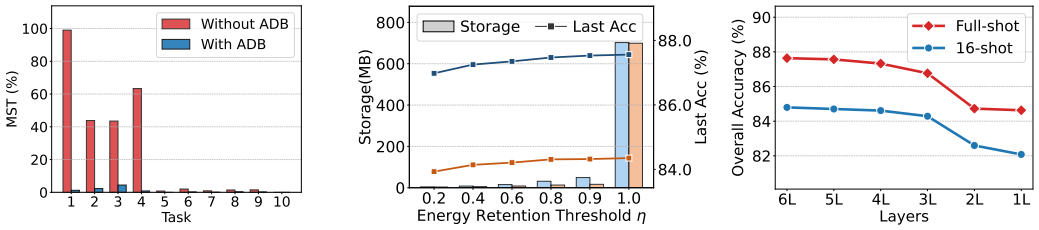
			16-shot			Full-shot		
OVA	**ADB**	**ARC**	**Transfer**	**Average**	**Last**	**Transfer**	**Average**	**Last**
✓	×	✓	62.8	72.5	78.8	62.7	74.2	81.9
✓	✓	×	63.4	73.1	82.3	63.1	75.8	86.6
×	✓	✓	63.3	74.0	84.1	63.1	76.3	87.2
✓	✓	✓	**63.4**	**74.4**	**84.4**	**63.1**	**76.5**	**87.5**

4.3 Ablation Study

To evaluate the contribution of each proposed component, we conduct an ablation study under the 16-shot and full-shot X-TAIL setting. We analyze three core modules: Orthogonal Visual Adaptation (OVA), Analytic Decision Boundary (ADB), and Analytic Ridge Classifier (ARC). We adopt a leave-one-out strategy to isolate the impact of each module. When ARC is omitted, standard CLIP cosine similarity is used for inference. Table 4 summarizes the results. As shown in Table 4, all three proposed components are effective and work synergistically across both the 16-shot and full-shot settings. Removing the Analytic Decision Boundary (ADB) results in a clear performance decrease. This indicates that explicit boundary calibration helps prevent semantic encroachment and mitigates forgetting. Omitting the Analytic Ridge Classifier (ARC) similarly reduces overall accuracy. Without ARC, the model loses fine-grained visual discriminability for seen domains, although zero-shot transfer remains stable since ARC exclusively targets seen-class predictions. Finally, excluding Orthogonal Visual Adaptation (OVA) leads to a consistent decline. This confirms that OVA effectively improves the visual encoder's adaptability to downstream datasets while mitigating visual representation degradation.

4.4 Analytical Study

Mitigating Asymmetric Boundary Encroachment via ADB. To isolate Asymmetric Boundary Encroachment from traditional representation degradation, we strictly freeze the image encoder.



(a) Misclassification (b) Storage-Accuracy Trade-off (c) Feature Depth Ablation

Figure 3: (a) Cross-task misclassification rates compared to the baseline. (b) Trade-off between storage overhead and model accuracy under varying energy retention thresholds. Blue and orange colors represent the results of full-shot and 16-shot settings respectively. (c) Overall accuracy across different numbers of evaluated layers.

Crucially, we fine-tune the text encoder for each task, extracting and strictly freezing its generated class embeddings. This guarantees zero historical representation degradation in both modalities. We then measure how often old-task images are erroneously classified into subsequent new tasks. As shown in Figure 3(a), where the x-axis represents the sequential task index (Task 1 to 10) and the y-axis denotes the misclassification rate (MST %), the baseline suffers severe Asymmetric Boundary Encroachment. For instance, error rates exceed 95% for Aircraft (Task 1) and 60% for Eurosat (Task 4). Enforcing ADB structurally bounds new text embeddings, effectively suppressing these errors to near-zero across all tasks.

Figure 4 explains this geometrically through the margin distributions $S_{c,c'} = \Delta \mathbf{w}_{c,c'}^T \mathbf{x}_c$. Baseline distributions incorrectly cross the zero-margin boundary ($S = 0$). Applying $\mathcal{L}_{\text{boundary}}$ shifts the distribution means to the right. This helps maintain a positive analytic 3σ lower bound, promoting a clearer separation of old class features from new text features.

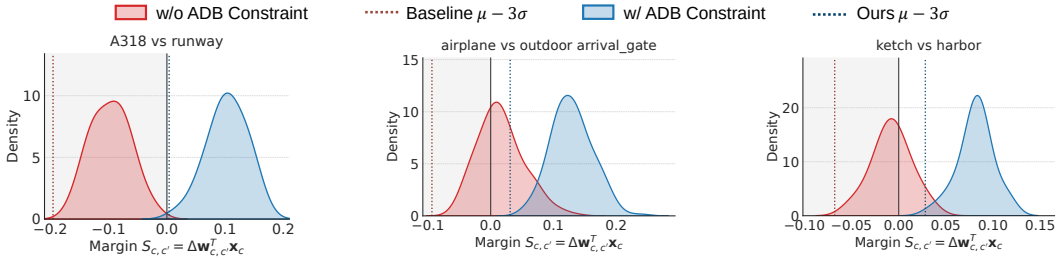


Figure 4: Margin distributions for three confusable class pairs. Unlike the baseline (red), whose left tail breaches the $S = 0$ decision boundary, our Analytic Decision Boundary (blue) shifts the distribution mean rightward. This helps maintain a positive analytic 3σ lower bound (dotted lines).

Trade-off Between Storage Overhead and Retained Variance. To justify our low-rank approximation of $\mathbf{\Sigma}_c$, Figure 3(b) evaluates the trade-off between accuracy and memory footprint. In this plot, the x-axis represents the energy retention threshold η, the left y-axis (bar plot) displays the storage overhead in MB, and the right y-axis (line plot) shows the final accuracy (Last Acc %). While higher retained variance monotonically improves accuracy by enriching the generator matrix $\mathbf{P}_c$, it incurs a highly nonlinear storage cost. Notably, increasing the threshold η from 0.9 to 1.0 (full rank) yields a marginal accuracy gain but triggers a memory surge from under 100 MB to approximately 700 MB. This sharp increase empirically validates our motivation to avoid the dense $\mathcal{O}(|\mathcal{C}|D^2)$ bottleneck. Consequently, we select $\eta = 0.9$ as the optimal threshold, effectively preserving representational fidelity while curbing memory expansion.

Effectiveness of Multi-Layer Statistics. We ablate the feature depth used for the Analytic Ridge Classifier (ARC) to validate the effectiveness of integrating multi-layer features. As shown in Figure 3(c) (x-axis: number of final layers accumulated, y-axis: overall accuracy %), relying solely on the final visual layer yields 82.0% (16-shot) and 84.5% (full-shot). Progressively incorporating intermediate statistics (Layer 11 back to 7) monotonically improves performance across both settings, plateauing at 84.4% and 87.5% respectively with 6 layers. This confirms that features from relatively shallower layers can effectively enhance inter-class discriminability.

5 Conclusion

In this work, we identify Asymmetric Boundary Encroachment (ABE) as an underexplored vulnerability in Vision-Language Models under the Cross-domain Task-Agnostic Incremental Learning (X-TAIL) setting, where new text features actively encroach upon established historical spaces. To systematically mitigate this, we propose a unified framework centered on the Analytic Decision Boundary (ADB), which establishes a geometric defense using cumulative feature statistics. By further integrating Orthogonal Visual Adaptation (OVA) and an Analytic Ridge Classifier (ARC), our approach safely evolves the visual backbone while maximizing joint inference capabilities. Experiments under the X-TAIL setting demonstrate that our framework effectively mitigates ABE and achieves new state-of-the-art performance.

6 Acknowledgement

This work was supported by the National Natural Science Foundation of China (62676119, 62236003), Shenzhen College Stability Support Plan (GXWD20231130151329002), Shenzhen Science and Technology Program (KQTD20240729102207002, KQTD2025120109171700), Guangdong Provincial Key Laboratory of Artificial General Intelligence and Large Models (2026B1212010005), and Guangdong Science and Technology Program (2025B0101130003).

References

[1] Kyra Ahrens, Hans Hergen Lehmann, Jae Hee Lee, and Stefan Wermter. Read between the layers: Leveraging multi-layer representations for rehearsal-free continual learning with pre-trained models. *Transactions on Machine Learning Research*, 2023.

[2] Lukas Bossard, Matthieu Guillaumin, and Luc Van Gool. Food-101–mining discriminative components with random forests. In *European conference on computer vision*, pages 446–461. Springer, 2014.

[3] Arslan Chaudhry, Marcus Rohrbach, Mohamed Elhoseiny, Thalaiyasingam Ajanthan, Puneet K Dokania, Philip HS Torr, and Marc'Aurelio Ranzato. On tiny episodic memories in continual learning. *arXiv preprint arXiv:1902.10486*, 2019.

[4] Mircea Cimpoi, Subhransu Maji, Iasonas Kokkinos, Sammy Mohamed, and Andrea Vedaldi. Describing textures in the wild. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3606–3613, 2014.

[5] Matthias De Lange, Rahaf Aljundi, Marc Masana, Sarah Parisot, Xu Jia, Aleš Leonardis, Gregory 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, 2021.

[6] Li Deng. The mnist database of handwritten digit images for machine learning research [best of the web]. *IEEE signal processing magazine*, 29(6):141–142, 2012.

[7] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

[8] Li Fei-Fei, Rob Fergus, and Pietro Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *2004 conference on computer vision and pattern recognition workshop*, pages 178–178. IEEE, 2004.

[9] Robert M French. Catastrophic forgetting in connectionist networks. *Trends in cognitive sciences*, 3(4):128–135, 1999.

[10] Peng Gao, Shijie Geng, Renrui Zhang, Teli Ma, Rongyao Fang, Yongfeng Zhang, Hongsheng Li, and Yu Qiao. Clip-adapter: Better vision-language models with feature adapters. *International journal of computer vision*, 132(2):581–595, 2024.

[11] Patrick Helber, Benjamin Bischke, Andreas Dengel, and Damian Borth. Eurosat: A novel dataset and deep learning benchmark for land use and land cover classification. *IEEE Journal of Selected Topics in Applied Earth Observations and Remote Sensing*, 12(7):2217–2226, 2019.

[12] Yen-Chang Hsu, Yen-Cheng Liu, Anita Ramasamy, and Zsolt Kira. Re-evaluating continual learning scenarios: A categorization and case for strong baselines. *arXiv preprint arXiv:1810.12488*, 2018.

[13] 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.

[14] Jonathan Krause, Michael Stark, Jia Deng, and Li Fei-Fei. 3d object representations for fine-grained categorization. In *Proceedings of the IEEE international conference on computer vision workshops*, pages 554–561, 2013.

[15] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017.

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

[17] Mao-Lin Luo, Zi-Hao Zhou, Tong Wei, and Min-Ling Zhang. LADA: Scalable label-specific CLIP adapter for continual learning. In *Forty-second International Conference on Machine Learning*, 2025.

[18] Subhransu Maji, Esa Rahtu, Juho Kannala, Matthew Blaschko, and Andrea Vedaldi. Fine-grained visual classification of aircraft. *arXiv preprint arXiv:1306.5151*, 2013.

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

[20] Michael McCloskey and Neal J Cohen. Catastrophic interference in connectionist networks: The sequential learning problem. In *Psychology of learning and motivation*, volume 24, pages 109–165. Elsevier, 1989.

[21] Mark D McDonnell, Dong Gong, Amin Parvaneh, Ehsan Abbasnejad, and Anton Van den Hengel. Ranpac: Random projections and pre-trained models for continual learning. *Advances in Neural Information Processing Systems*, 36:12022–12053, 2023.

[22] Maria-Elena Nilsback and Andrew Zisserman. Automated flower classification over a large number of classes. In *2008 Sixth Indian conference on computer vision, graphics & image processing*, pages 722–729. IEEE, 2008.

[23] Feiyang Ning and Xinyang Chen. Mitigating endogenous confirmation bias in noisy label learning for vision-language models. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pages 24576–24584, 2026.

[24] Aristeidis Panos, Yuriko Kobe, Daniel Olmeda Reino, Rahaf Aljundi, and Richard E Turner. First session adaptation: A strong replay-free baseline for class-incremental learning. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 18820–18830, 2023.

[25] Omkar M Parkhi, Andrea Vedaldi, Andrew Zisserman, and CV Jawahar. Cats and dogs. In *2012 IEEE conference on computer vision and pattern recognition*, pages 3498–3505. IEEE, 2012.

[26] Tiantian Peng, Yuyang Liu, Shuo Yang, Qiuhe Hong, and YongHong Tian. Gnsp: Gradient null space projection for preserving cross-modal alignment in vlms continual learning. *arXiv preprint arXiv:2507.19839*, 2025.

[27] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International conference on machine learning*, pages 8748–8763. PmLR, 2021.

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

[29] Gobinda Saha, Isha Garg, and Kaushik Roy. Gradient projection memory for continual learning. In *International Conference on Learning Representations*, 2021.

[30] Hanul Shin, Jung Kwon Lee, Jaehong Kim, and Jiwon Kim. Continual learning with deep generative replay. *Advances in neural information processing systems*, 30, 2017.

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

[32] Longxiang Tang, Zhuotao Tian, Kai Li, Chunming He, Hantao Zhou, Hengshuang Zhao, Xiu Li, and Jiaya Jia. Mind the interference: Retaining pre-trained knowledge in parameter efficient continual learning of vision-language models. In *European conference on computer vision*, pages 346–365. Springer, 2024.

[33] Liyuan Wang, Xingxing Zhang, Hang Su, and Jun Zhu. A comprehensive survey of continual learning: Theory, method and application. *IEEE transactions on pattern analysis and machine intelligence*, 46(8):5362–5383, 2024.

[34] Yuchen Wang, Xuefeng Bai, Xiucheng Li, Weili Guan, Liqiang Nie, and Xinyang Chen. Handling imbalanced pseudolabels for vision-language models with concept alignment and confusion-aware calibrated margin. In *Proceedings of the 42nd International Conference on Machine Learning*, volume 267, pages 62357–62372, 2025.

[35] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 139–149, 2022.

[36] Mitchell Wortsman, Gabriel Ilharco, Jong Wook Kim, Mike Li, Simon Kornblith, Rebecca Roelofs, Raphael Gontijo Lopes, Hannaneh Hajishirzi, Ali Farhadi, Hongseok Namkoong, et al. Robust fine-tuning of zero-shot models. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 7959–7971, 2022.

[37] Jiayang Wu, Xinyang Chen, Ke Lv, and Weili Guan. Boosting visual reprogramming for clip with dual granularity alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 29347–29356, 2026.

[38] Jianxiong Xiao, James Hays, Krista A Ehinger, Aude Oliva, and Antonio Torralba. Sun database: Large-scale scene recognition from abbey to zoo. In *2010 IEEE computer society conference on computer vision and pattern recognition*, pages 3485–3492. IEEE, 2010.

[39] Yicheng Xu, Yuxin Chen, Jiahao Nie, Yusong Wang, Huiping Zhuang, and Manabu Okumura. Advancing cross-domain discriminability in continual learning of vision-language models. *Advances in Neural Information Processing Systems*, 37:51552–51576, 2024.

[40] Jiazuo Yu, Yunzhi Zhuge, Lu Zhang, Ping Hu, Dong Wang, Huchuan Lu, and You He. Boosting continual learning of vision-language models via mixture-of-experts adapters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23219–23230, 2024.

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

[42] Renrui Zhang, Wei Zhang, Rongyao Fang, Peng Gao, Kunchang Li, Jifeng Dai, Yu Qiao, and Hongsheng Li. Tip-adapter: Training-free adaption of clip for few-shot classification. In *European conference on computer vision*, pages 493–510. Springer, 2022.

[43] Zhen Zhao, Zhizhong Zhang, Xin Tan, Jun Liu, Yanyun Qu, Yuan Xie, and Lizhuang Ma. Rethinking gradient projection continual learning: Stability / plasticity feature space decoupling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 3718–3727, 2023.

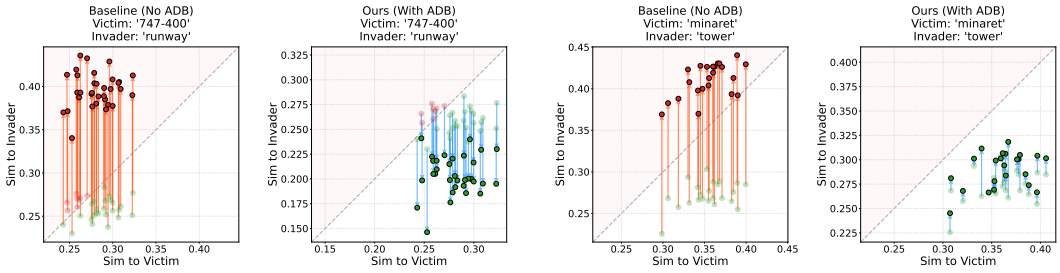
[44] Zangwei Zheng, Mingyuan Ma, Kai Wang, Ziheng Qin, Xiangyu Yue, and Yang You. Preventing zero-shot transfer degradation in continual learning of vision-language models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 19125–19136, 2023.

[45] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Conditional prompt learning for vision-language models. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022.

[46] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision (IJCV)*, 2022.

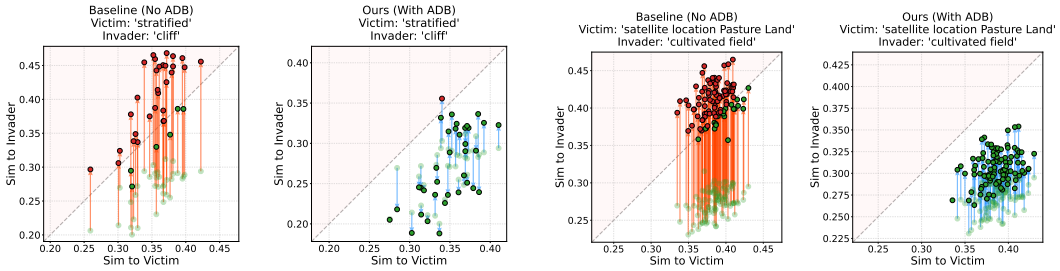
Appendix

A Trajectory Analysis: Unconstrained Encroachment Amidst Stable Historical Representations



(a) Sun397 encroaching upon Aircraft

(b) Sun397 encroaching upon Caltech101



(c) Sun397 encroaching upon DTD

(d) Sun397 encroaching upon EuroSAT

Figure 5: Visualization of Asymmetric Boundary Encroachment and the Defensive Efficacy of ADB.

To validate the mechanics of Asymmetric Boundary Encroachment, we visualize the similarity trajectories of historical image samples (Victims) transitioning from Step $T-1$ to T in Fig. 5. In our experimental setup, visual representations are constrained by the Orthogonal Visual Adaptation (OVA) component to prevent representation degradation. For the text modality, the text encoder is trained on the current dataset, after which the learned text features are cached and frozen. The X-axis represents the similarity between these historical image features and their cached text, while the Y-axis denotes their similarity to the newly learned "Invader" text.

In the Baseline, the trajectories exhibit a striking vertical shift. The stable X-axis coordinates demonstrate that the text-image similarity within historical classes remains highly consistent, suggesting that, in this controlled setting, the observed errors are not primarily caused by collapse of original alignments. Instead, misclassification is driven exclusively by a sharp Y-axis surge: unconstrained new text features actively increase their similarity to historical images, dragging previously safe instances across the decision boundary. This directly visualizes Asymmetric Boundary Encroachment. Conversely, our Analytic Decision Boundary (ADB) enforces a geometric defense that successfully reduces this optimization-driven vertical encroachment, tightly confining the similarity trajectories within the correct classification zone.

B Pervasive Encroachment: Optimization-Driven Dynamics Beyond High-Similarity Pairs

To analyze the scope of Asymmetric Boundary Encroachment, Figure 6 illustrates the relationship between encroachment and the text feature similarity of new and historical classes, alongside the specific severity of encroachment across individual classes when learning the new task 'Sun397'. As in our setup, visual representations are constrained by the Orthogonal Visual Adaptation (OVA) component to prevent representation degradation. For the text modality, the text encoder is trained on the current dataset, after which the learned text features are cached and frozen. In the scatter plots, the

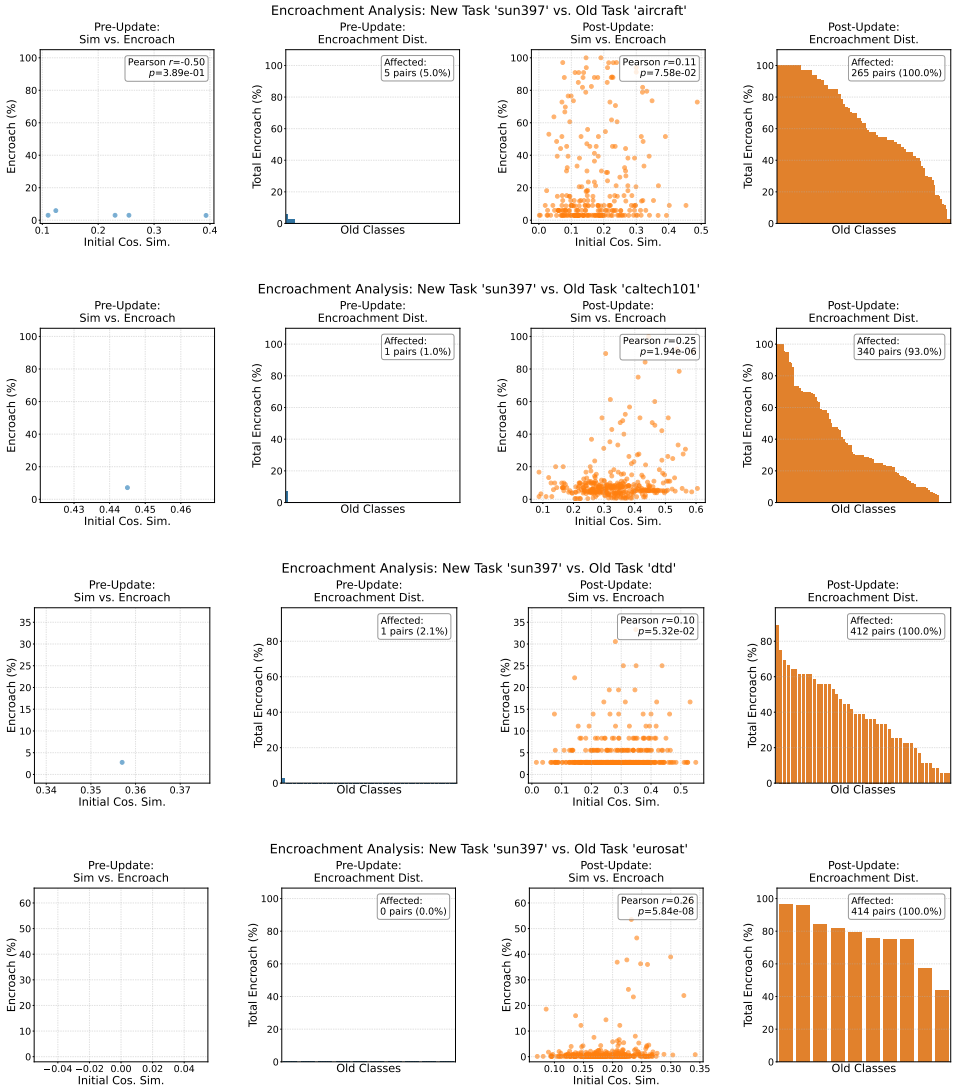


Figure 6: Empirical analysis reveals Asymmetric Boundary Encroachment as a widespread, optimization-driven process not limited to highly similar classes.

X-axis measures the cosine similarity between trained historical text features and original new text features, representing the similarity landscape before optimizing the new task. The Y-axis tracks the encroach rate, calculated as the proportion of historical samples misclassified into the new category. Adjacent bar charts depict total encroachment across historical classes. Comparing the Pre-Update and Post-Update states reveals key mechanics. First, the initial absence of encroachment confirms this is an optimization-driven process rather than static feature overlap. Second, the post-update bar charts demonstrate a pervasive encroachment across numerous old classes. Furthermore, the low Pearson correlation ($r \leq 0.26$) in the scatter plots indicates this phenomenon is not restricted to highly similar class pairs but spans a broad spectrum of categories. This widespread vulnerability underlines the necessity of a geometric defense like ADB to secure established multi-modal boundaries.

C Decoupling Asymmetric Boundary Encroachment from Representation Degradation via Margin Decomposition

To analytically trace the drivers of catastrophic forgetting, we define the relative margin $M(x; f, g)$ for a specific old-class sample x misclassified into a new class c_{new}, with its ground-truth class being

c_{old}. Let f and g represent the visual and text encoders:

$$M(x; f, g) = \cos(f(x), g(c_{new})) - \cos(f(x), g(c_{old})) \tag{13}$$

Using subscripts 0 and 1 to denote the states before and after learning the new task, the margin shift $\Delta M(x)$ driving the misclassification of sample x is:

$$\Delta M(x) = M(x; f_1, g_1) - M(x; f_0, g_0) \tag{14}$$

By inserting an intermediate state $M(x; f_0, g_1)$, we decouple this shift into two distinct, sample-specific components: the Encroachment Margin (EM) and the Degradation Margin (DM):

$$EM(x) = M(x; f_0, g_1) - M(x; f_0, g_0) \tag{15}$$

$$DM(x) = M(x; f_1, g_1) - M(x; f_0, g_1) \tag{16}$$

such that $\Delta M(x) = EM(x) + DM(x)$.

Specifically, $EM(x)$ anchors the visual features at $f_0(x)$ to purely measure how text-space updates compress the decision boundary. It encapsulates the asymmetric boundary encroachment, accounting for the misclassification pull from text-side variations. Conversely, $DM(x)$ isolates how visual backbone drift (from f_0 to f_1) disrupts multimodal alignment, assuming a finalized text space g_1.

To evaluate the overall impact, we quantify the text-driven Encroachment Dominance Score (EDS) over the subset of misclassified samples $\mathcal{D}_{err}$ (where ϵ is a small constant for numerical stability):

$$EDS(c_{new}, \mathcal{D}_{err}) = \mathbb{E}_{x \in \mathcal{D}_{err}} \left[\frac{|EM(x)|}{|EM(x)| + |DM(x)| + \epsilon} \right] \tag{17}$$

For our experiments, we analyze the margin shift on old datasets before and after learning the new SUN397 dataset. As shown in Table 5, the baseline reveals that the misclassification of old-class samples is not a single-factor failure, but the joint result of visual representation degradation and semantic boundary encroachment. When OVA constrains the visual features, optimization pressure shifts to the unconstrained text modality, triggering a severe compensatory surge in semantic boundary encroachment ($EDS \approx 1$). Finally, our synergistic strategy (OVA combined with ADB) effectively suppresses this misclassification dynamic, dropping the ER to a remarkably low level.

Table 5: Decomposition of margin shift across different configurations before and after training on SUN397. ER (%) denotes the rate of old-dataset samples misclassified into SUN397.

Configuration	**Dataset**	**EDS**	**ER (%)**
No OVA, No ADB	Aircraft	0.0999	94.69
	Caltech101	0.3726	29.01
	DTD	0.6503	36.70
	EuroSAT	0.7437	37.48
With OVA, No ADB	Aircraft	0.9996	64.96
	Caltech101	0.9997	39.72
	DTD	0.9997	37.41
	EuroSAT	0.9998	76.43
With OVA, With ADB	Aircraft	0.9872	0.90
	Caltech101	0.9965	0.73
	DTD	0.9974	1.54
	EuroSAT	0.9995	0.42

D Detailed Evaluation Metrics

Following the X-TAIL protocol [39], we evaluate our framework across a sequence of K tasks. Let $A_{k,j}$ denote the classification accuracy on domain j after completing the training for task k. The three evaluation metrics are formally defined as follows:

- **Transfer:** Measures the model's capacity to maintain the pre-trained Vision-Language Model's zero-shot generalization. It is computed as the average zero-shot accuracy on unseen domains before they are sequentially learned:

$$\mathcal{M}_{\text{transfer}} = \frac{1}{K-1} \sum_{k=1}^{K-1} \left(\frac{1}{K-k} \sum_{j=k+1}^{K} A_{k,j} \right) \tag{18}$$

- **Average:** Evaluates the holistic stability and performance of the model throughout the continuous learning trajectory. It calculates the mean accuracy across all domains (both seen and unseen) evaluated at every incremental learning step:

$$\mathcal{M}_{\text{average}} = \frac{1}{K} \sum_{k=1}^{K} \left(\frac{1}{K} \sum_{j=1}^{K} A_{k,j} \right) \tag{19}$$

- **Last:** Reflects the ultimate retention of incrementally acquired knowledge after the model has completed the training sequence for all K tasks:

$$\mathcal{M}_{\text{last}} = \frac{1}{K} \sum_{j=1}^{K} A_{K,j} \tag{20}$$

E Implementation Details

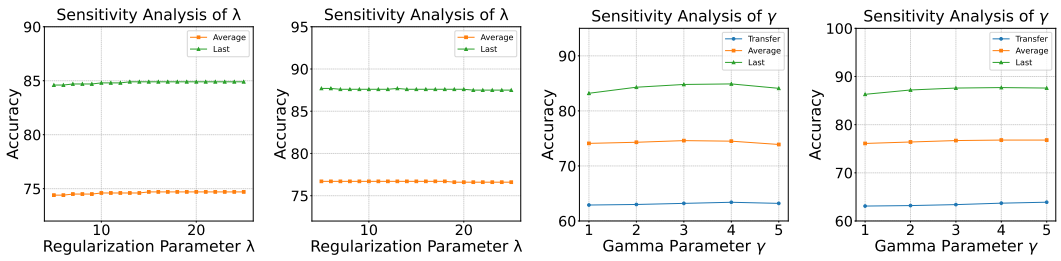
We implement our method using PyTorch based on the CLIP[27] model of the ViT-B/16 image encoder[7]. We follow the dataset setting adopted in CoOp [45, 46], including the class definitions, and train/test splits. All experiments are conducted on a single NVIDIA A100 (40GB) GPU. We employ the AdamW optimizer with a batch size of 128, an initial learning rate of 0.001, and a weight decay of 5×10^{-4}, utilizing the OneCycleLR scheduler. During the adaptation to downstream tasks, we employ the standard cross-entropy loss between the current task's input images and the textual labels of all seen classes as the primary classification objective. The probabilistic margin multiplier γ (Section 3.2) is set to 3, the energy retention threshold η for dynamic eigendecomposition truncation (Section 3.2) is set to 0.9, and the ridge regression penalty for inference is set to $\lambda_{\text{ridge}} = 10$ (Section 3.3). For ARC (Section 3.3), we form the regularized Gram matrix $G_{\text{reg}} = G + \lambda I$ and solve the ridge system in practice using `torch.linalg.solve` on GPU.

We evaluate framework robustness through the sensitivity of the ARC penalty λ and the ADB margin multiplier γ. As shown in Fig. 7 (a-b), performance remains stable across $\lambda \in [5, 25]$ in both 16-shot and full-shot settings, demonstrating our framework's robustness to the choice of the ARC penalty. Meanwhile, varying γ (Fig. 7 (c-d)) reveals optimal accuracy at $\gamma = 3$ or 4. This validates our probabilistic design: looser bounds fail to prevent Asymmetric Boundary Encroachment, while excessively strict bounds harm model plasticity. Overall, these trends confirm the practical reliability of our analytic constraints.

F Computational Efficiency Comparison with Representative Methods

To demonstrate the computational efficiency of our proposed framework, we compare its peak GPU memory footprint, total trainable parameters, and average training time per batch against representative methods, including ZSCL [44], MoE-Adapters [40], and LADA [17]. The measurements are conducted under identical hardware configurations (a single NVIDIA A100 40GB GPU) with a batch size of 64, consistent with the experimental settings reported in the LADA [17] paper to ensure a fair comparison.

As shown in Table 6, our method reduces computational overhead compared to these methods. Specifically, our framework requires approximately only half of the trainable parameters of LADA (4.54 M vs. 9.01 M), while also consuming less peak memory (12.94 GB vs. 18.02 GB) and achieving a shorter average training time (0.245s vs. 0.280s). Compared to ZSCL and MoE-Adapters, our method further demonstrates substantial reductions in trainable parameters, peak memory consumption, and training time. Furthermore, RAIL [39] freezes CLIP and requires zero trainable parameters, but its



(a) 16-shot (λ) (b) full-shot (λ) (c) 16-shot (γ) (d) full-shot (γ)

Figure 7: Sensitivity analysis of the hyperparameters in our framework under 16-shot and full-shot settings. (a, b) The regularization parameter λ used in the Analytic Ridge Classifier (ARC) to stabilize the inverse of the accumulated Gram matrix. (c, d) The probabilistic margin multiplier γ (corresponding to γ in Eq. 3 and Eq. 7) used in the Analytic Decision Boundary (ADB) to mitigate Asymmetric Boundary Encroachment.

analytic learning paradigm introduces additional storage and computational overhead. Specifically, Primal-RAIL employs a 15,000-dimensional projection requiring approximately 900 MB of storage, while Dual-RAIL stores all raw features and encounters GPU out-of-memory (OOM) issues and computational overhead under the full-shot setting. These results demonstrate the resource efficiency of our framework in continual learning.

Table 6: Comparison of computational efficiency between representative continual learning methods and our proposed method. All metrics are evaluated using a single NVIDIA A100 GPU with a batch size of 64. The units used are millions (M) for trainable parameters, gigabytes (GB) for peak memory, and seconds (s) for training time.

Method	**Params (M)**	**Peak Memory (GB)**	**Time / Batch (s)**
ZSCL [44]	149.62	30.49	1.362
MoE-Adapters [40]	51.06	24.77	0.608
LADA [17]	9.01	18.02	0.280
Ours	**4.54**	**12.94**	**0.245**

G Computational Complexity and Scalability of the Analytic Decision Boundary

To demonstrate the scalability of the Analytic Decision Boundary (ADB) as the number of historical classes $|\mathcal{C}_{\text{old}}|$ increases, we analyze its asymptotic and empirical computational costs.

Asymptotic Complexity Derivation. Evaluating the geometric margin constraints requires calculating the boundary violations between all historical classes and the incoming new classes of the current task, denoted as $|\mathcal{C}_{\text{new}}|$. For each of the $|\mathcal{C}_{\text{old}}| \cdot |\mathcal{C}_{\text{new}}|$ class pairs, calculating the margin variance involves projecting the weight differential vector $\Delta\mathbf{w}_{c,c'} \in \mathbb{R}^D$ onto the class-specific low-rank generator matrix $\mathbf{P}_c \in \mathbb{R}^{D\times M}$. Here, D is the underlying feature dimension of the model, and M is the retained low-rank dimension ($M \ll D$). As empirically validated in Table 7, M consistently constitutes only a small fraction of D across diverse datasets. This specific matrix-vector multiplication requires $\mathcal{O}(M \cdot D)$ floating-point operations. Consequently, the total asymptotic complexity to compute the geometric boundary across all pairs per step is strictly bounded by:

$$\mathcal{O}(|\mathcal{C}_{\text{old}}| \cdot |\mathcal{C}_{\text{new}}| \cdot M \cdot D) \tag{21}$$

Crucially, this operation acts solely on class-level cached statistics and text classifier weights. It is therefore independent of the batch size B (the number of image-text pairs processed per forward step) and the sequence length L (the number of input tokens per sample, such as image patches in ViT or sub-word tokens in the text encoder). To contextualize the efficiency of ADB, we can contrast it with the fundamental computational cost of the vision-language Transformer backbone. During the standard feature extraction pass, dense linear projections (e.g., query/key/value mappings and

feed-forward networks) demand $\mathcal{O}(D^2)$ operations per token, yielding $\mathcal{O}(B \cdot L \cdot D^2)$ operations globally. Additionally, the self-attention mechanism, which computes pairwise similarities across the entire token sequence, incurs a cost of $\mathcal{O}(B \cdot L^2 \cdot D)$. Thus, the base backbone scales as:

$$\mathcal{O}(B \cdot L \cdot D^2 + B \cdot L^2 \cdot D) \tag{22}$$

Given that $M \ll D$ and the boundary loss entirely circumvents the computationally dense B and L dimensions, the theoretical overhead introduced by ADB is expected to remain modest relative to the base feature extraction.

Table 7: Empirical statistics of the retained low-rank dimension M across 10 domains. With the energy retention threshold η set to 0.9, M is dynamically determined and constitutes only a small fraction of the original feature dimension $D = 512$.

Dataset	**Classes**	**Full Shot**		**16-Shot**	
		Avg M	**Min - Max**	**Avg** M	**Min - Max**
Aircraft	100	20.73	16 - 23	11.28	10 - 12
Caltech101	100	20.65	10 - 86	10.92	9 - 12
DTD	47	31.43	27 - 35	10.94	10 - 12
EuroSAT	10	59.80	29 - 74	9.40	6 - 11
Flowers	102	21.30	12 - 48	10.84	10 - 12
Food101	101	101.84	89 - 127	11.55	11 - 12
MNIST	10	42.10	35 - 61	8.00	6 - 10
Oxford Pets	37	39.81	35 - 43	11.16	10 - 12
Stanford Cars	196	20.65	13 - 30	11.09	10 - 12
SUN397	397	24.29	18 - 29	11.24	10 - 13

Empirical Implementation and Vectorization. The double summation $\sum_{c \in \mathcal{C}_{\text{old}}} \sum_{c' \in \mathcal{C}_{\text{new}}}$ associated with the pairwise constraints is not executed via explicit Python loops. Instead, we leverage tensor broadcasting to compute the pairwise weight differentials $\Delta \mathbf{w}_{c,c'}$ simultaneously. The subsequent projections onto the low-rank generator matrices $\mathbf{P}_c$ are executed via a single parallelized batch matrix multiplication (torch.bmm). This vectorization reduces the constraint evaluation to exceptionally fast, GPU-accelerated dense linear algebra operations.

Wall-clock Overhead and Scaling Behavior. To validate the real-world efficiency of ADB, we profiled its exact execution time during the 10-domain sequential continual learning process. All measurements were conducted under the identical training configuration detailed in Section E. Accurately measuring the execution of GPU-accelerated operations is challenging due to the asynchronous nature of PyTorch and CPU-bound data loading bottlenecks. To ensure precision and isolate the pure GPU computational cost of the ADB loss, we wrapped the specific operations using asynchronous CUDA timing events (`torch.cuda.Event(enable_timing=True)`). We enforced explicit device synchronization (`torch.cuda.synchronize()`) at the end of each batch to ensure all asynchronous kernels were completed before recording the elapsed time. The profiling results, averaged across training steps, are presented in Table 8.

As shown in Table 8, for the initial domain (Aircraft), no historical classes exist ($|\mathcal{C}_{\text{old}}| = 0$); hence, the ADB loss computation is naturally bypassed, yielding zero overhead. For subsequent tasks, the empirical measurements align with our asymptotic derivation $\mathcal{O}(|\mathcal{C}_{\text{old}}| \cdot |\mathcal{C}_{\text{new}}| \cdot M \cdot D)$. Notably, for datasets with a small number of incoming classes (e.g., EuroSAT and MNIST where $|\mathcal{C}_{\text{new}}| = 10$), the ADB execution time actually drops, despite the continuous growth of $|\mathcal{C}_{\text{old}}|$. Even in the final and most extreme scenario tested (SUN397), where the model evaluates boundary violations across a massive 703×397 class pairs simultaneously, the absolute calculation time remains under 5 milliseconds per batch. Across the entire incremental sequence, the relative wall-clock overhead introduced by ADB is consistently bounded below 1.5%, solidifying its scalability for continual learning applications.

H Limitations

Our framework achieves strong performance on the X-TAIL benchmark. A limitation is that although the Analytic Decision Boundary (ADB) is computationally efficient in practice, its computational cost is not constant: at each incremental step, it must evaluate boundary constraints between every old class and every incoming new class, so the cost grows with the number of old-new class pairs.

Table 8: Empirical wall-clock overhead per training step during the 10-domain sequential continual learning process. Time measurements isolate true GPU execution using synchronized CUDA events.

Incoming Task	$\lvert\mathcal{C}_{\text{old}}\rvert$	$\lvert\mathcal{C}_{\text{new}}\rvert$	**Execution Time per Batch**		
			ADB Time (ms)	**Total Time (ms)**	**Overhead (%)**
Aircraft	0	100	0.000	198.834	0.000%
Caltech101	100	100	2.462	186.505	1.320%
DTD	200	47	0.896	188.976	0.474%
EuroSAT	247	10	0.547	149.812	0.365%
Flowers	257	102	1.070	197.770	0.541%
Food101	359	101	0.799	160.591	0.498%
MNIST	460	10	0.530	143.868	0.368%
Oxford Pets	470	37	1.273	197.719	0.644%
Stanford Cars	507	196	3.073	222.085	1.384%
SUN397	703	397	4.615	324.908	1.421%

Table 9: Comparison of different CL methods on X-TAIL 16-shot for each task with order-II in terms of "Transfer", "Average", and "Last" scores (%). Results for our method and LADA are averaged over 5 random seeds. The best results are highlighted in **bold**.

	Method	Cars	Aircraft	Pets	Food101	SUN397	MNIST	Flowers	DTD	Caltech101	EuroSAT	**Average**
	Zero-shot	65.5	23.8	87.8	83.4	60.8	43.9	64.1	36.4	74.3	37.4	57.7
Transfer	LwF [15]	–	20.0	74.1	79.6	58.1	34.1	48.9	27.7	64.4	15.1	46.9
	WiSE-FT [36]	–	21.3	79.5	83.3	61.0	39.9	56.5	29.6	68.0	20.8	51.1
	ZSCL [44]	–	23.0	84.3	87.2	**63.0**	42.1	65.2	34.6	71.4	**40.9**	56.9
	MoE-Adapters [40]	–	17.1	87.2	**87.5**	58.4	12.6	65.5	35.9	70.0	17.9	50.2
	LADA [17]	–	**23.8**	87.8	84.3	61.3	40.7	65.7	**36.7**	71.5	34.7	56.3
	Ours	–	**23.8**	**88.1**	84.6	62.0	**49.3**	**66.3**	36.1	**72.8**	40.1	**58.1**
Average	LwF [15]	49.0	27.4	69.7	83.0	65.7	42.2	63.5	33.1	68.5	17.5	52.0
	WiSE-FT [36]	57.9	29.6	77.8	85.4	68.0	51.6	69.3	35.5	71.0	23.0	56.9
	ZSCL [44]	74.4	36.4	86.7	**88.7**	68.9	50.0	75.1	40.1	72.5	43.7	63.6
	MoE-Adapters [40]	74.4	38.6	87.7	87.3	67.9	50.6	76.5	43.7	72.3	18.8	61.8
	Primal-RAIL [39]	81.1	43.0	91.1	86.6	69.0	59.2	76.7	45.6	**78.4**	42.3	67.3
	Dual-RAIL [39]	82.3	43.9	91.0	85.6	69.3	59.5	77.1	45.8	**78.4**	42.5	67.5
	LADA [17]	84.8	46.1	92.4	86.1	70.1	66.6	78.1	46.2	75.4	40.0	68.6
	Ours	**86.2**	**49.8**	**92.4**	86.4	**71.1**	**72.5**	**79.1**	**46.3**	77.3	**45.2**	**70.6**
Last	LwF [15]	29.6	17.5	63.0	83.8	67.7	44.9	79.3	44.8	84.6	39.0	55.4
	WiSE-FT [36]	46.1	23.5	71.3	85.7	70.2	59.1	85.5	47.9	82.4	42.8	61.5
	ZSCL [44]	71.7	35.3	86.5	**89.2**	71.8	52.3	89.8	52.0	77.1	68.4	69.4
	MoE-Adapters [40]	75.1	41.1	87.9	87.1	74.1	89.7	92.6	61.2	81.0	27.4	71.7
	Primal-RAIL [39]	80.8	45.0	92.3	86.7	75.1	91.8	96.4	68.2	94.6	86.9	81.8
	Dual-RAIL [39]	82.3	46.3	92.2	86.9	75.7	92.6	97.4	69.0	94.7	88.3	82.5
	LADA [17]	84.9	49.5	**93.7**	87.2	76.7	92.6	97.5	68.5	91.4	87.4	82.9
	Ours	**85.9**	**51.9**	93.6	87.2	**77.2**	**95.5**	**98.3**	**70.1**	**95.6**	**90.4**	**84.6**

I Comparison of Different Methods on X-TAIL with Order-II

In this section, we compare various Continual Learning (CL) methods within the X-TAIL 16-shot and full-shot settings using an alternative random task sequence (Order-II): Cars, Aircraft, Pets, Food101, SUN397, MNIST, Flowers, DTD, Caltech101, and EuroSAT. As demonstrated in Table 9 and Table 10, our proposed method consistently outperforms previous approaches across all key metrics (Transfer, Average, and Last). These findings demonstrate the robustness of our approach to different task orderings.

J Mean and Standard Deviation over Random Seeds

The results of Ours and LADA are reported as the mean over five random seeds. We report the per-task mean and standard deviation (mean $\pm$ std, in %) for both order-I (Table 11) and order-II

Table 10: Comparison of different CL methods on X-TAIL full-shot for each task with order-II in terms of "Transfer", "Average", and "Last" scores (%). Results for our method and LADA are averaged over 5 random seeds. The best results are highlighted in **bold**.

	Method	Cars	Aircraft	Pets	Food101	SUN397	MNIST	Flowers	DTD	Caltech101	EuroSAT	Average
	Zero-shot	65.5	**23.8**	87.8	83.4	60.8	43.9	64.1	36.4	74.3	37.4	57.7
Transfer	LwF [15]	–	16.7	78.5	76.0	59.7	41.3	46.6	27.3	63.3	10.4	46.6
	WiSE-FT [36]	–	20.3	77.9	75.7	55.6	39.6	45.0	25.4	58.9	8.3	45.2
	ZSCL [44]	–	21.7	83.2	85.6	**63.0**	39.3	61.8	34.3	**72.2**	26.4	54.2
	MoE-Adapters [40]	–	17.5	87.1	**86.8**	58.2	44.2	63.4	33.9	67.9	15.3	52.7
	LADA [17]	–	**23.8**	87.8	84.3	61.4	40.4	65.7	33.9	70.2	27.2	55.0
	Ours	–	23.5	**88.2**	84.6	62.7	**48.7**	**66.3**	**34.8**	70.9	**34.7**	**57.1**
Average	LwF [15]	38.6	21.6	73.2	75.9	67.7	69.4	62.2	36.8	68.0	15.1	52.9
	WiSE-FT [36]	47.1	30.9	77.9	76.6	65.0	59.0	58.7	36.1	65.4	9.8	52.7
	ZSCL [44]	77.8	43.1	90.3	**89.5**	71.8	61.3	73.6	42.9	74.2	26.8	65.1
	MoE-Adapters [40]	84.2	47.4	89.0	88.0	65.2	70.7	76.3	43.4	71.5	14.8	65.1
	Primal-RAIL [39]	83.0	45.7	92.1	87.1	70.0	61.6	76.8	46.4	**78.3**	**43.1**	68.4
	LADA [17]	87.2	48.9	**93.1**	87.9	71.0	67.6	78.9	46.5	75.3	34.1	69.0
	Ours	**89.0**	**54.4**	**93.1**	87.7	**72.4**	**73.5**	**79.5**	**47.8**	76.1	40.8	**71.4**
Last	LwF [15]	13.4	6.9	58.2	74.5	71.2	**99.4**	76.3	53.2	85.4	57.0	59.6
	WiSE-FT [36]	19.3	2.0	70.5	77.6	67.9	72.1	66.2	52.7	90.6	22.7	54.2
	ZSCL [44]	75.6	31.7	90.4	**90.7**	77.0	75.4	88.3	60.5	82.1	29.6	70.1
	MoE-Adapters [40]	84.1	50.6	88.9	88.1	68.7	97.2	95.5	65.6	86.2	10.3	73.5
	Primal-RAIL [39]	81.9	46.1	93.3	89.0	76.6	98.2	96.6	70.6	94.1	94.1	84.1
	LADA [17]	87.3	55.3	**94.6**	89.6	77.7	98.8	98.9	75.9	95.9	**95.8**	87.0
	Ours	**88.7**	**57.0**	94.4	89.1	**78.7**	98.3	**99.2**	**78.2**	**96.6**	**95.8**	**87.6**

Table 11: Per-task mean and standard deviation (%) over 5 random seeds for order-I. The rightmost column reports the overall mean across tasks.

	Method	air.	cal.	dtd.	eur.	flo.	foo.	mni.	oxf.	sta.	sun.	Average
Transfer	LADA (16-shot)	–	$75.0_{\pm 0.0}$	$36.2_{\pm 0.2}$	$36.2_{\pm 0.8}$	$66.5_{\pm 0.1}$	$84.1_{\pm 0.2}$	$40.1_{\pm 1.4}$	$88.0_{\pm 0.0}$	$65.4_{\pm 0.1}$	$61.6_{\pm 0.1}$	$61.5_{\pm 0.2}$
	Ours (16-shot)	–	$76.6_{\pm 0.1}$	$36.7_{\pm 0.1}$	$42.2_{\pm 0.3}$	$65.3_{\pm 0.5}$	$84.1_{\pm 0.2}$	$49.1_{\pm 1.7}$	$88.4_{\pm 0.2}$	$65.5_{\pm 0.1}$	$62.2_{\pm 0.1}$	$63.4_{\pm 0.2}$
	LADA (full-shot)	–	$75.3_{\pm 0.1}$	$35.9_{\pm 0.1}$	$36.7_{\pm 0.2}$	$65.9_{\pm 0.2}$	$84.0_{\pm 0.1}$	$44.6_{\pm 1.7}$	$88.0_{\pm 0.0}$	$65.4_{\pm 0.0}$	$61.2_{\pm 0.1}$	$61.9_{\pm 0.2}$
	Ours (full-shot)	–	$76.6_{\pm 0.2}$	$36.3_{\pm 0.1}$	$42.3_{\pm 0.4}$	$64.6_{\pm 0.2}$	$83.8_{\pm 0.2}$	$48.1_{\pm 0.9}$	$88.4_{\pm 0.1}$	$65.7_{\pm 0.1}$	$62.0_{\pm 0.2}$	$63.1_{\pm 0.2}$
Average	LADA (16-shot)	$48.8_{\pm 0.7}$	$89.2_{\pm 1.0}$	$60.7_{\pm 0.7}$	$72.0_{\pm 0.3}$	$84.6_{\pm 0.1}$	$85.2_{\pm 0.2}$	$61.1_{\pm 1.1}$	$89.6_{\pm 0.1}$	$69.3_{\pm 0.1}$	$63.0_{\pm 0.1}$	$72.4_{\pm 0.2}$
	Ours (16-shot)	$53.2_{\pm 0.8}$	$92.0_{\pm 0.3}$	$61.3_{\pm 1.1}$	$75.6_{\pm 0.3}$	$85.2_{\pm 0.3}$	$85.6_{\pm 0.1}$	$67.7_{\pm 1.0}$	$90.0_{\pm 0.1}$	$69.5_{\pm 0.1}$	$63.8_{\pm 0.1}$	$74.4_{\pm 0.2}$
	LADA (full-shot)	$53.1_{\pm 0.9}$	$93.7_{\pm 0.4}$	$67.0_{\pm 0.2}$	$77.9_{\pm 0.1}$	$85.4_{\pm 0.1}$	$86.7_{\pm 0.0}$	$64.8_{\pm 0.4}$	$89.9_{\pm 0.0}$	$69.8_{\pm 0.1}$	$62.8_{\pm 0.1}$	$75.1_{\pm 0.1}$
	Ours (full-shot)	$59.7_{\pm 0.3}$	$93.2_{\pm 0.2}$	$68.3_{\pm 0.4}$	$79.9_{\pm 0.1}$	$85.4_{\pm 0.1}$	$86.5_{\pm 0.2}$	$68.2_{\pm 0.5}$	$90.3_{\pm 0.1}$	$70.2_{\pm 0.1}$	$63.6_{\pm 0.2}$	$76.5_{\pm 0.1}$
Last	LADA (16-shot)	$49.4_{\pm 0.6}$	$93.2_{\pm 0.5}$	$69.1_{\pm 0.8}$	$87.3_{\pm 0.5}$	$97.0_{\pm 0.2}$	$87.1_{\pm 0.1}$	$92.6_{\pm 1.3}$	$93.6_{\pm 0.4}$	$84.8_{\pm 0.6}$	$76.3_{\pm 0.2}$	$83.0_{\pm 0.2}$
	Ours (16-shot)	$52.9_{\pm 0.8}$	$94.7_{\pm 0.4}$	$69.3_{\pm 1.4}$	$89.9_{\pm 0.5}$	$98.4_{\pm 0.3}$	$87.2_{\pm 0.2}$	$95.6_{\pm 0.7}$	$93.6_{\pm 0.7}$	$85.5_{\pm 0.4}$	$77.5_{\pm 0.1}$	$84.4_{\pm 0.2}$
	LADA (full-shot)	$55.3_{\pm 0.5}$	$96.2_{\pm 0.2}$	$76.1_{\pm 0.2}$	$95.7_{\pm 0.1}$	$98.4_{\pm 0.2}$	$89.6_{\pm 0.1}$	$98.8_{\pm 0.0}$	$94.6_{\pm 0.1}$	$87.5_{\pm 0.2}$	$77.3_{\pm 0.1}$	$86.9_{\pm 0.1}$
	Ours (full-shot)	$58.9_{\pm 0.3}$	$95.7_{\pm 0.2}$	$77.1_{\pm 0.1}$	$95.6_{\pm 0.1}$	$99.2_{\pm 0.1}$	$89.0_{\pm 0.4}$	$98.3_{\pm 0.0}$	$94.7_{\pm 0.1}$	$88.0_{\pm 0.1}$	$78.6_{\pm 0.1}$	$87.5_{\pm 0.1}$

(Table 12) under the full-shot and 16-shot settings. For compact presentation, both the mean and standard deviation are rounded to one decimal place.

K Statistical Significance Analysis Against the State-of-the-Art

To rigorously validate that the performance improvements of our method over the strongest baseline (LADA) are not due to random variance across different seeds, we conducted a formal statistical significance analysis.

We employed the Paired t-test to compare the raw performance scores of our method and LADA across 5 identical random seeds. The analysis was performed comprehensively on four experimental settings: Order-I (16-shot), Order-I (full-shot), Order-II (16-shot), and Order-II (full-shot). We evaluate the significance across all three core metrics: Transfer, Average, and Last.

As summarized in Table 13, the calculated p-values across all settings and metrics are consistently and strictly below the rigorous threshold of $p < 0.01$ (ranging from 10^{-3} to 10^{-6}). These results suggest that the observed improvements over LADA are statistically reliable across random seeds.

Table 12: Per-task mean and standard deviation (%) over 5 random seeds for order-II. The rightmost column reports the overall mean across tasks.

	Method	sta.	air.	oxf.	foo.	sun.	mni.	flo.	dtd.	cal.	eur.	Average
Transfer	LADA (16-shot)	–	$23.8_{\pm 0.0}$	$87.8_{\pm 0.0}$	$84.3_{\pm 0.0}$	$61.3_{\pm 0.0}$	$40.7_{\pm 3.0}$	$65.7_{\pm 0.0}$	$36.7_{\pm 0.5}$	$71.5_{\pm 1.0}$	$34.7_{\pm 2.3}$	$56.3_{\pm 0.7}$
	Ours (16-shot)	–	$23.8_{\pm 0.3}$	$88.1_{\pm 0.1}$	$84.6_{\pm 0.1}$	$62.0_{\pm 0.2}$	$49.3_{\pm 0.9}$	$66.3_{\pm 0.3}$	$36.1_{\pm 0.3}$	$72.8_{\pm 0.3}$	$40.1_{\pm 1.8}$	$58.1_{\pm 0.2}$
	LADA (full-shot)	–	$23.8_{\pm 0.0}$	$87.8_{\pm 0.0}$	$84.3_{\pm 0.0}$	$61.4_{\pm 0.0}$	$40.4_{\pm 1.7}$	$65.7_{\pm 0.0}$	$33.9_{\pm 0.6}$	$70.2_{\pm 0.6}$	$27.2_{\pm 1.4}$	$55.0_{\pm 0.3}$
	Ours (full-shot)	–	$23.5_{\pm 0.2}$	$88.2_{\pm 0.2}$	$84.6_{\pm 0.1}$	$62.7_{\pm 0.2}$	$48.7_{\pm 1.2}$	$66.3_{\pm 0.3}$	$34.8_{\pm 0.4}$	$70.9_{\pm 0.4}$	$34.7_{\pm 1.5}$	$57.1_{\pm 0.0}$
Average	LADA (16-shot)	$84.8_{\pm 0.3}$	$46.1_{\pm 0.3}$	$92.4_{\pm 0.4}$	$86.1_{\pm 0.1}$	$70.1_{\pm 0.1}$	$66.6_{\pm 1.3}$	$78.1_{\pm 0.4}$	$46.2_{\pm 0.7}$	$75.4_{\pm 0.7}$	$40.0_{\pm 2.0}$	$68.6_{\pm 0.4}$
	Ours (16-shot)	$86.2_{\pm 0.4}$	$49.8_{\pm 0.5}$	$92.4_{\pm 0.5}$	$86.4_{\pm 0.2}$	$71.1_{\pm 0.0}$	$72.5_{\pm 0.3}$	$79.1_{\pm 0.2}$	$46.3_{\pm 0.5}$	$77.3_{\pm 0.3}$	$45.2_{\pm 1.6}$	$70.6_{\pm 0.2}$
	LADA (full-shot)	$87.2_{\pm 0.3}$	$48.9_{\pm 0.8}$	$93.1_{\pm 0.1}$	$87.9_{\pm 0.0}$	$71.0_{\pm 0.1}$	$67.6_{\pm 0.6}$	$78.9_{\pm 0.1}$	$46.5_{\pm 0.4}$	$75.3_{\pm 0.5}$	$34.1_{\pm 1.2}$	$69.0_{\pm 0.1}$
	Ours (full-shot)	$89.0_{\pm 0.1}$	$54.4_{\pm 0.4}$	$93.1_{\pm 0.2}$	$87.7_{\pm 0.0}$	$72.4_{\pm 0.1}$	$73.5_{\pm 0.6}$	$79.5_{\pm 0.2}$	$47.8_{\pm 0.3}$	$76.1_{\pm 0.3}$	$40.8_{\pm 1.4}$	$71.4_{\pm 0.1}$
Last	LADA (16-shot)	$84.9_{\pm 0.3}$	$49.5_{\pm 0.7}$	$93.7_{\pm 0.4}$	$87.2_{\pm 0.3}$	$76.7_{\pm 0.2}$	$92.6_{\pm 1.2}$	$97.5_{\pm 0.1}$	$68.5_{\pm 1.7}$	$91.4_{\pm 0.8}$	$87.4_{\pm 0.6}$	$82.9_{\pm 0.2}$
	Ours (16-shot)	$85.9_{\pm 0.5}$	$51.9_{\pm 0.6}$	$93.6_{\pm 0.8}$	$87.2_{\pm 0.2}$	$77.2_{\pm 0.1}$	$95.5_{\pm 0.6}$	$98.3_{\pm 0.2}$	$70.1_{\pm 1.0}$	$95.6_{\pm 0.2}$	$90.4_{\pm 0.5}$	$84.6_{\pm 0.2}$
	LADA (full-shot)	$87.3_{\pm 0.3}$	$55.3_{\pm 0.3}$	$94.6_{\pm 0.2}$	$89.6_{\pm 0.0}$	$77.7_{\pm 0.1}$	$98.8_{\pm 0.0}$	$98.9_{\pm 0.2}$	$75.9_{\pm 0.3}$	$95.9_{\pm 0.2}$	$95.8_{\pm 0.1}$	$87.0_{\pm 0.0}$
	Ours (full-shot)	$88.7_{\pm 0.1}$	$57.0_{\pm 0.6}$	$94.4_{\pm 0.3}$	$89.1_{\pm 0.0}$	$78.7_{\pm 0.1}$	$98.3_{\pm 0.1}$	$99.2_{\pm 0.1}$	$78.2_{\pm 0.3}$	$96.6_{\pm 0.1}$	$95.8_{\pm 0.0}$	$87.6_{\pm 0.1}$

Table 13: Statistical significance analysis (Paired t-test) comparing our method against LADA across 5 random seeds. All reported p-values are far below the standard threshold of 0.01, indicating that our performance improvements are highly statistically significant.

Setting	Metric	Mean Score		p-value	Significance
		LADA	Ours		
Order-I (16-shot)	Transfer	61.46	**63.37**	6.88×10^{-5}	Significant ($p < 0.01$)
	Average	72.36	**74.38**	1.96×10^{-5}	Significant ($p < 0.01$)
	Last	83.04	**84.45**	9.21×10^{-5}	Significant ($p < 0.01$)
Order-I (Full-shot)	Transfer	61.88	**63.09**	7.36×10^{-4}	Significant ($p < 0.01$)
	Average	75.10	**76.52**	3.22×10^{-6}	Significant ($p < 0.01$)
	Last	86.94	**87.52**	4.79×10^{-4}	Significant ($p < 0.01$)
Order-II (16-shot)	Transfer	56.29	**58.13**	2.43×10^{-3}	Significant ($p < 0.01$)
	Average	68.60	**70.63**	2.40×10^{-4}	Significant ($p < 0.01$)
	Last	82.94	**84.58**	1.80×10^{-5}	Significant ($p < 0.01$)
Order-II (Full-shot)	Transfer	54.96	**57.15**	8.22×10^{-5}	Significant ($p < 0.01$)
	Average	69.05	**71.42**	2.38×10^{-6}	Significant ($p < 0.01$)
	Last	86.97	**87.61**	2.18×10^{-4}	Significant ($p < 0.01$)

L Detailed Performance

In this section, we provide the detailed, per-training-step accuracies of our proposed method on the X-TAIL benchmark. Specifically, we present the comprehensive result matrices for both the 16-shot and full-shot settings, evaluated under two distinct task sequences: Order-I and Order-II (see Tables 14 through 17). The detailed results show that our method consistently integrates new domain knowledge while preserving previously acquired knowledge across all experimental settings.

Table 14: Accuracy of ours on the X-TAIL 16-shot with order-I. The diagonal values (shaded) represent the performance immediately after training on the respective task. *Transfer*, *Average*, and *Last* metrics are highlighted with a light background, and the final aggregated results are in **bold**. All reported values represent the mean across five random seeds.

Task	Aircraft	Caltech101	DTD	EuroSAT	Flowers	Food	MNIST	Pets	Cars	SUN397	
Transfer	-	76.6	36.7	42.2	65.3	84.1	49.1	88.4	65.5	62.2	**63.4**
Aircraft	54.1	76.6	36.6	42.6	64.2	83.7	49.6	88.1	65.6	61.5	
Caltech101	53.9	93.4	36.8	42.6	65.8	84.1	49.6	88.5	65.5	61.7	
DTD	53.9	93.4	66.6	41.6	65.7	84.3	50.8	88.5	65.5	61.8	
EuroSAT	53.4	93.3	66.2	90.0	65.7	84.3	48.3	88.5	65.5	62.5	
Flowers	53.1	93.4	66.5	89.9	98.5	84.3	48.1	88.5	65.5	62.5	
Food	52.9	93.4	67.9	90.0	98.5	87.2	48.1	88.5	65.5	62.5	
MNIST	52.7	93.3	67.6	89.8	98.4	87.2	95.5	88.5	65.5	62.5	
Pets	52.6	93.5	67.7	89.7	98.4	87.2	95.5	93.6	65.5	62.5	
Cars	52.7	94.7	67.6	89.8	98.4	87.1	95.5	93.6	85.6	62.6	
SUN397 (*Last*)	52.9	94.7	69.3	89.9	98.4	87.2	95.6	93.6	85.5	77.5	**84.4**
Average	53.2	92.0	61.3	75.6	85.2	85.6	67.7	90.0	69.5	63.8	**74.4**

Table 15: Accuracy of ours on the X-TAIL full-shot with order-I. The diagonal values (shaded) represent the performance immediately after training on the respective task. *Transfer*, *Average*, and *Last* metrics are highlighted with a light background, and the final aggregated results are in **bold**. All reported values represent the mean across five random seeds.

Task	Aircraft	Caltech101	DTD	EuroSAT	Flowers	Food	MNIST	Pets	Cars	SUN397	
Transfer	-	76.6	36.3	42.3	64.6	83.8	48.1	88.4	65.7	62.0	**63.1**
Aircraft	61.2	76.6	36.3	42.6	64.4	83.7	49.9	88.2	65.7	61.4	
Caltech101	60.7	94.7	36.3	42.6	64.7	83.9	50.1	88.5	65.7	61.6	
DTD	60.0	94.9	76.0	41.8	64.6	83.9	51.9	88.4	65.7	61.5	
EuroSAT	60.0	94.8	75.8	96.2	64.6	83.9	45.7	88.4	65.7	62.2	
Flowers	59.6	94.9	76.1	96.1	99.4	83.9	45.6	88.4	65.7	62.3	
Food	59.3	94.9	76.4	96.0	99.3	89.1	45.5	88.4	65.7	62.2	
MNIST	59.3	94.8	76.3	95.9	99.3	89.2	98.4	88.4	65.7	62.2	
Pets	59.1	94.8	76.3	95.9	99.3	89.2	98.3	94.8	65.7	62.1	
Cars	58.9	95.7	76.2	95.8	99.2	89.2	98.4	94.8	88.2	62.2	
SUN397 (*Last*)	58.9	95.7	77.1	95.6	99.2	89.0	98.3	94.7	88.0	78.6	**87.5**
Average	59.7	93.2	68.3	79.9	85.4	86.5	68.2	90.3	70.2	63.6	**76.5**

Table 16: Accuracy of ours on the X-TAIL 16-shot with order-II. The diagonal values (shaded) represent the performance immediately after training on the respective task. *Transfer*, *Average*, and *Last* metrics are highlighted with a light background, and the final aggregated results are in **bold**. All reported values represent the mean across five random seeds.

Task	Cars	Aircraft	Pets	Food	SUN397	MNIST	Flowers	DTD	Caltech101	EuroSAT	
Transfer	-	23.8	88.1	84.6	62.0	49.3	66.3	36.1	72.8	40.1	**58.1**
Cars	86.5	23.8	88.1	84.6	62.0	49.5	66.4	36.3	73.4	42.7	
Aircraft	86.6	53.6	88.1	84.6	62.1	49.5	66.4	36.3	74.0	42.8	
Pets	86.6	53.4	93.4	84.6	62.0	49.3	66.3	36.2	74.0	42.8	
Food	86.4	53.1	93.4	87.0	62.0	49.2	66.3	36.6	74.1	42.8	
SUN397	86.2	52.9	93.3	87.1	77.3	49.2	66.3	35.7	71.6	39.1	
MNIST	86.2	52.8	93.4	87.1	77.2	95.8	66.3	35.7	71.6	38.3	
Flowers	86.1	52.5	93.4	87.1	77.2	95.7	98.5	35.7	72.0	38.3	
DTD	86.1	52.4	93.4	87.2	77.1	95.7	98.4	70.4	71.3	37.6	
Caltech101	85.9	52.0	93.6	87.2	77.1	95.6	98.3	70.0	95.6	37.0	
EuroSAT (*Last*)	85.9	51.9	93.6	87.2	77.2	95.5	98.3	70.1	95.6	90.4	**84.6**
Average	86.2	49.8	92.4	86.4	71.1	72.5	79.1	46.3	77.3	45.2	**70.6**

Table 17: Accuracy of ours on the X-TAIL full-shot with order-II. The diagonal values (shaded) represent the performance immediately after training on the respective task. *Transfer*, *Average*, and *Last* metrics are highlighted with a light background, and the final aggregated results are in **bold**. All reported values represent the mean across five random seeds.

Task	Cars	Aircraft	Pets	Food	SUN397	MNIST	Flowers	DTD	Caltech101	EuroSAT	
Transfer	-	23.5	88.2	84.6	62.7	48.7	66.3	34.8	70.9	34.7	**57.1**
Cars	89.6	23.5	88.2	84.6	62.5	49.4	66.3	36.6	73.7	42.8	
Aircraft	89.5	58.9	88.2	84.6	62.7	49.5	66.3	36.5	74.2	42.9	
Pets	89.4	58.5	94.4	84.6	62.7	49.2	66.3	36.6	74.3	42.8	
Food	89.1	58.1	94.4	89.0	62.7	48.9	66.3	36.2	74.2	42.8	
SUN397	88.8	58.0	94.1	88.9	78.9	46.4	66.3	33.3	69.1	29.6	
MNIST	88.9	57.7	94.1	89.1	79.0	98.4	66.3	32.8	69.1	28.0	
Flowers	88.8	57.5	94.2	89.1	78.9	98.4	99.3	31.4	68.5	28.0	
DTD	88.8	57.3	94.1	89.1	78.8	98.4	99.2	78.7	64.5	27.8	
Caltech101	88.7	57.2	94.4	89.1	78.7	98.3	99.2	78.2	96.5	27.5	
EuroSAT (*Last*)	88.7	57.0	94.4	89.1	78.7	98.3	99.2	78.2	96.6	95.8	**87.6**
Average	89.0	54.4	93.1	87.7	72.4	73.5	79.5	47.8	76.1	40.8	**71.4**

NeurIPS Paper Checklist

1. **Claims**

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

   Answer: [Yes]

   Justification: The main claims in the introduction are consistent with the proposed method and the reported results; see Sections 1, 3, and 4.

   Guidelines:

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

2. **Limitations**

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

   Answer: [Yes]

   Justification: The limitations of the proposed method are discussed in Appendix Section H.

   Guidelines:

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

3. **Theory assumptions and proofs**

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

   Answer: [N/A]

Justification:The paper does not present theorem-style theoretical results requiring formal proofs.

Guidelines:

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

4. **Experimental result reproducibility**

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

   Answer:[Yes]

   Justification: The paper provides the benchmark protocol, evaluation metrics, implementation details, and detailed appendix results needed to reproduce the main findings; see Sections 4.1, E, D, and I.

   Guidelines:

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

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

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

Answer: [Yes]

Justification: The paper provides an anonymized public code repository in the abstract, and the repository and supplemental material include the instructions needed to reproduce the main experimental results.

Guidelines:

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

6. **Experimental setting/details**

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

   Answer: [Yes]

   Justification: The manuscript specifies the benchmark, metrics, optimizer, batch size, scheduler, and key hyperparameters; see Sections 4.1 and E.

   Guidelines:

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

7. **Experiment statistical significance**

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

   Answer: [Yes]

   Justification: The main results are averaged over 5 random seeds, and the appendix includes paired t-tests with p-values; see Section J and Section K.

   Guidelines:

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

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

8. **Experiments compute resources**

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

   Answer: [Yes]

   Justification: The paper reports the hardware platform, GPU memory usage, and execution time statistics relevant for reproducing the experiments; see Sections E, F, and G.

   Guidelines:

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

9. **Code of ethics**

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

   Answer: [Yes]

   Justification: To the best of our knowledge, this work complies with the NeurIPS Code of Ethics. The study is methodological and does not involve human subjects or crowdsourcing; see Sections 3 and 4.

   Guidelines:

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

10. **Broader impacts**

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

    Answer: [N/A]

    Justification: This paper focuses on a general methodological problem and does not target a specific deployment setting with direct societal impact.

Guidelines:

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

11. **Safeguards**

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

    Answer: [N/A]

    Justification: The paper does not release a high-risk model or scraped dataset requiring special safeguards. The submission focuses on method development and benchmark evaluation; see Sections 3 and 4.

    Guidelines:

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

12. **Licenses for existing assets**

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

    Answer: [Yes]

    Justification: All the datasets and models are properly cited.

    Guidelines:

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

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

13. **New assets**

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

    Answer: [N/A]

    Justification: The submission does not release new asset.

    Guidelines:

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

14. **Crowdsourcing and research with human subjects**

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

    Answer: [N/A]

    Justification: The paper does not involve crowdsourcing or human-subject research.

    Guidelines:

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

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

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

    Answer: [N/A]

    Justification: The paper does not involve human subjects or crowdsourcing.

    Guidelines:

    - The answer [N/A] means that the paper does not involve crowdsourcing nor research with human subjects.

- Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
- We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
- For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

    Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does *not* impact the core methodology, scientific rigor, or originality of the research, declaration is not required.

    Answer: [N/A]

    Justification: The LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology.

    Guidelines:

    - The answer [N/A] means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
    - Please refer to our LLM policy in the NeurIPS handbook for what should or should not be described.