INNERCIL: INTERNAL NEURON REPRESENTATIONS FOR ANALYTIC CLASS-INCREMENTAL LEARNING

Anonymous authors
Paper under double-blind review

ABSTRACT

Class-incremental learning aims to enable a model to sequentially learn new classes while preserving its ability to recognize old ones. Recently, analytic learning methods have achieved advanced performance by updating classifiers via accumulated features, with the pretrained backbone frozen. However, existing approaches typically focus on transforming or resampling the output features during analysis, where transforming and resampling can introduce feature drift and feature noise respectively, limiting the robustness of the obtained classifier. To address this problem, we propose InnerCIL, where the Internal Neuron Representations are exploited to provide informative and stable information for classifier analysis. InnerCIL consists of two key components, Attention-conditioned Native Dictionary Generation (ANDG) and Fisher Refinement Analytic Classification (FRAC). Specifically, instead of merely utilizing output features, ANDG introduces a Neuron-guided attention mechanism to extract a set of local representations within middle layers. These features form a native dictionary where no additional transform is adopted. Then, based on the dictionary containing local and global features of different stages, FRAC filters a subset to accelerate classifier analysis while mitigating the noise samples that can misguide the classifier. As a result, InnerCIL is both effective and efficient. Experiments on different backbones and benchmarks show that our method outperforms existing methods by a large margin.

1 INTRODUCTION

Class-Incremental Learning (CIL) aims to learn a sequence of tasks while retaining previously acquired knowledge. Its central challenge is catastrophic forgetting, where the model's performance on previously learned classes degrades severely as it continuously learns new classes.

Existing CIL methods primarily update the entire network or a subset of its parameters, as shown in Fig. 1 (a). However, continuously updating the model parameters inevitably causes representation drift, which can lead to severe catastrophic forgetting. Recently, analytic CIL methods (Zhuang et al., 2022) have demonstrated promising performance and efficiency compared with conventional trainable CIL methods. These methods typically employ a fixed pre-trained model (PTM) as the feature extractor and fit the classifier in closed form. However, existing analytic CIL methods often rely solely on the global output of the feature extractor and introduce random nonlinear projections (McDonnell et al., 2023) or auxiliary lightweight networks (Momeni et al., 2025) to augment the features for classifier construction. As illustrated in Fig. 1 (b), these feature transformations may introduce confusing samples into the feature space, thereby misleading the decision boundary.

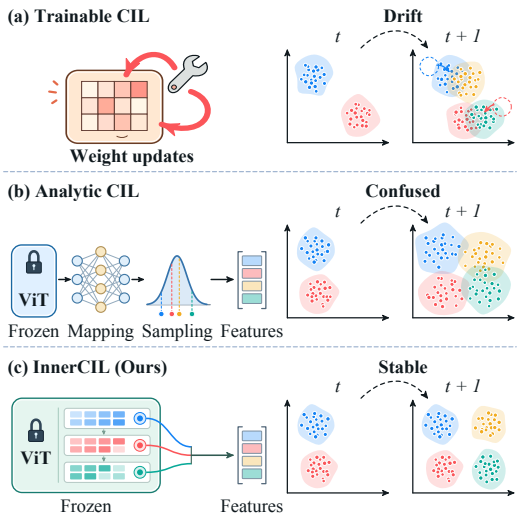


Figure 1: Conceptual comparison of different CIL paradigms.

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

To address these limitations, we propose InnerCIL, an analytic CIL framework that constructs the classifier based on the internal neuron responses of the feature extractor. Our key insight is that each pretrained neuron implicitly evaluates a nonlinear function over contextualized patch representations. The resulting response map records where the neuron is activated before the block's output projection mixes responses across neurons. Therefore, extracting these internal neuron responses can provide richer information than relying solely on the output features and their variants.

Specifically, an Attention-conditioned Native Dictionary Generation (ANDG) mechanism is developed, which employs CLS-to-patch attention to aggregate each response map into an image-level coordinate. Each coordinate preserves the identity of its pretrained neuron while emphasizing patches that receive greater attention from the model. By extracting such coordinates at multiple depths, ANDG constructs a dictionary of pretrained functions at patch and neuron resolution, complemented by multi-depth CLS features that provide global contextual information. A checkpoint-only registration rule bounds the candidate dictionary, while the response maps and pooling weights are obtained within the same frozen forward pass. Furthermore, we propose a Fisher Refinement Analytic Classification (FRAC) scheme, which uses cumulative class statistics to select local coordinates with strong inter-class separation and low intra-class variation. These selected coordinates complement the global branch within a joint ridge classifier. By maintaining sufficient statistics, the proposed method reconstructs the corresponding batch ridge solution as new classes arrive, thereby preserving analytic closure under a strict replay-free protocol. Extensive experiments across diverse CIL benchmarks demonstrate that InnerCIL is both effective and efficient compared with state-of-the-art methods.

The contributions of this paper are three-fold: (1) We identify the limited representational capacity of existing analytic CIL methods during classifier construction, and introduce a novel framework InnerCIL, which exploits internal native representations from pretrained models to enable more precise classifier construction. (2) An Attention-conditioned Native Dictionary Generation mechanism is developed to extract informative internal representations based on neuron responses. Besides, a Fisher Refinement Analytic Classification scheme is proposed, which actively filters the noisy features to improve the robustness of the obtained classifier. (3) Extensive experiments across diverse CIL datasets and pretrained backbones demonstrate that InnerCIL achieves state-of-the-art performance.

2 Related Work

Class-Incremental Learning. Recent class-incremental learning methods address forgetting through data replay, parameter regularization, dynamic network expansion, knowledge distillation, model rectification, and template-based classification (Zhou et al., 2024). A key distinction is the information retained from earlier tasks. Under a strict replay-free protocol, past knowledge must be preserved through model parameters or aggregate statistics, without storing old examples or generating synthetic ones. With strong pretrained models, the focus increasingly shifts from learning representations from scratch to adapting or exploiting transferable representations.

CIL with Pretrained Models. PTM-based CIL follows three main strategies: fine-tuning the backbone or lightweight adapters, learning prompts while freezing the backbone, and using the PTM as a fixed feature extractor with a prototype or analytic classifier. SLCA (Zhang et al., 2023) and InfLoRA (Liang & Li, 2024) represent the first strategy. L2P (Wang et al., 2022b), DualPrompt (Wang et al., 2022a), and CODA-Prompt (Smith et al., 2023) represent the second, which also includes predictive prompting in PrePrompt (Huang et al., 2026), while SimpleCIL (Zhou et al., 2025), APER (Zhou et al., 2025), and FeCAM (Goswami et al., 2023) belong to the third. MiN (Jiang et al., 2025) studies the use of noise mixtures with pretrained representations. CLIP-based methods additionally exploit language supervision through representation adjustment, modality-gap compensation, bridge-layer fusion, or attribute aggregation, as exemplified by RAPF (Huang et al., 2024), MG-CLIP (Huang et al., 2025), and BOFA (Li et al., 2026). BOFA is particularly relevant because it reaches into CLIP's internal knowledge-bridge layer rather than relying only on the model's output embeddings.

Analytic CIL. Analytic methods update classifiers from compact statistics using closed-form rules. NCM (Mensink et al., 2013) and SLDA (Hayes & Kanan, 2020) accumulate class moments. ACIL (Zhuang et al., 2022) and GACL (Zhuang et al., 2024) use recursive regularized least squares, APER combines a frozen and a first-session-adapted representation, and FeCAM models class-dependent covariance geometry.

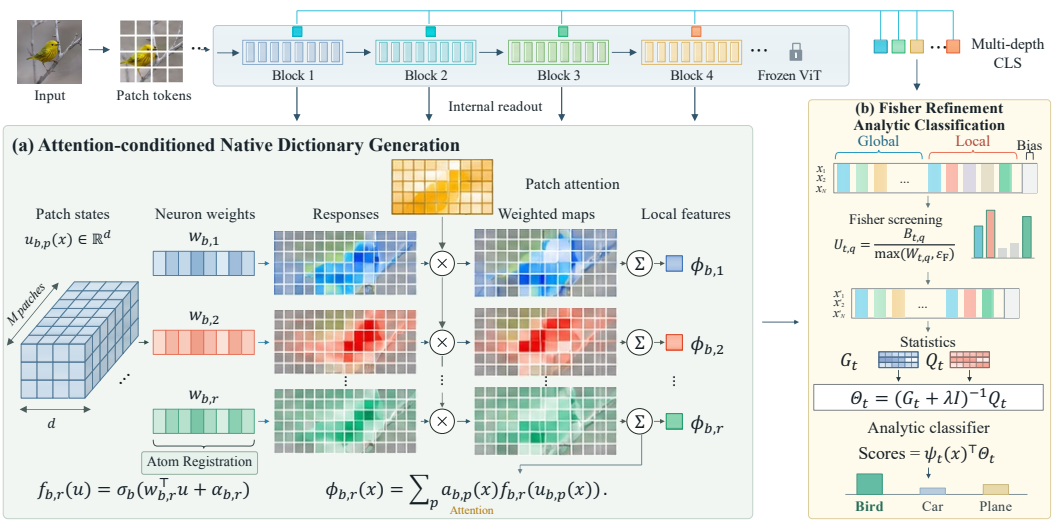


Figure 2: Overview of InnerCIL. CLS-to-patch attention pools pretrained neuron responses into local image coordinates, complemented by multi-depth CLS features. Both branches reuse a single frozen forward pass, and selected coordinates form a feature matrix for analytic classification from cumulative sufficient statistics.

RanPAC (McDonnell et al., 2023) is the closest methodological starting point of InnerCIL, augmenting final PTM representations with fixed random nonlinear features and an analytic classifier. LoRanPAC (Peng et al., 2024) improves this system through continual truncation, while InnerCIL retains its fixed-function principle but draws capacity from native internal responses. LayUP (Ahrens et al., 2023) forms coarse-grained multi-layer prototypes from concatenated intermediate representations. InnerCIL instead resolves fine-grained patch- and neuron-level responses through attention-conditioned functions. AnaCP (Momeni et al., 2025) learns analytic contrastive projections and fits an ELM-style classifier using sampled class-conditioned features. InnerCIL uses no gradient-optimized representation parameters and remains strictly replay-free.

3 Methodology

3.1 Preliminaries

Class-Incremental Learning. Class-incremental learning (CIL) receives a sequence of class-disjoint training sets $\mathcal{D}_t = \{(x_{t,n}, y_{t,n})\}_{n=1}^{n_t}$, with labels in $\mathcal{C}_t$. After task t, a single classifier recognizes all classes in $\mathcal{C}_{1:t} := \bigcup_{\tau \le t} \mathcal{C}_\tau$ without access to the task identity. Let $C_t = |\mathcal{C}_{1:t}|$ and let $\mathbf{e}_y \in \mathbb{R}^{C_t}$ denote the one-hot code of class y. Under strict exemplar-free CIL (EF-CIL), learning at task t accesses only $\mathcal{D}_t$: no old image or per-sample old feature is retained, and no synthetic old sample is generated or used. Class-level aggregates and additive sufficient statistics may persist across tasks.

Analytic Class-Incremental Learning. Analytic CIL (ACIL) formulates the incremental classifier as ridge regression over a fixed feature map $\psi : \mathcal{X} \to \mathbb{R}^D$. Let X_t contain the current-task feature rows and $\mathbf{Y}_t$ their one-hot targets.

$$\Theta_t := \arg\min_{\Theta} \left\| \mathbf{Y}_{1:t} - X_{1:t}\Theta \right\|_F^2 + \lambda \|\Theta\|_F^2, \qquad \lambda > 0. \tag{1}$$

Writing $G_t := X_{1:t}^\top X_{1:t}$ and $Q_t := X_{1:t}^\top \mathbf{Y}_{1:t}$, differentiating with respect to Θ yields $\Theta_t = (G_t + \lambda I)^{-1} Q_t$.

3.2 Overview

Figure 2 summarizes InnerCIL. Its central representation is a native global–local dictionary constructed directly from a frozen ViT. Multi-depth CLS tokens provide global coordinates, while CLS-to-patch attention aggregates checkpoint MLP-neuron responses into local image coordinates.

Checkpoint registration bounds the candidate capacity, and cumulative Visual Fisher selects a stage-dependent support. InnerCIL learns a unified analytic classifier that combines global context with local neuron responses, determining its weights directly from accumulated sufficient statistics through a closed-form ridge solution, without backpropagation or iterative parameter optimization.

3.3 Attention-Conditioned Native Dictionary Generation

An Inside View of ViT. Each hidden unit of a frozen ViT MLP is a nonlinear scalar function. Evaluating it on every patch produces a spatial response map, which CLS-to-patch attention converts into an image-level coordinate. This retains two resolutions of the frozen computation: global CLS summaries and fine-grained patch responses. Each local coordinate preserves the identity of one pretrained neuron before the MLP output projection mixes it with other hidden responses.

We consider frozen ViTs with standard two-layer MLP blocks. The backbone remains in deterministic evaluation mode throughout the sequence. Let the ViT contain L transformer blocks, and let $\mathcal{B} \subseteq \{0, \dots, L-1\}$ denote a subset of these blocks, with $L_{\mathcal{B}} := |\mathcal{B}| \geq 1$. The associated native-neuron universe is $\mathcal{Q} := \{(b, r) : b \in \mathcal{B}, 1 \leq r \leq m_b\}$. For block $b \in \mathcal{B}$, let $s_{b,p}(x) \in \mathbb{R}^d$ be token p after the attention residual and before the second LayerNorm. Here, $p = 0$ denotes CLS and $p \in \{1, \dots, M\}$ denotes a patch. The checkpoint MLP is

$$\mathrm{MLP}_b(u) = W_{2,b}\sigma_b(W_{1,b}u + c_{1,b}) + c_{2,b}, \quad W_{1,b} \in \mathbb{R}^{m_b \times d}, \quad W_{2,b} \in \mathbb{R}^{d \times m_b}. \tag{2}$$

At token p, set $u_{b,p}(x) := \mathrm{LN}_{2,b}(s_{b,p}(x))$. Native coordinate r is $f_{b,r}(u) := \sigma_b(w_{b,r}^\top u + \alpha_{b,r})$, where $w_{b,r} := W_{1,b}[r, :]^\top$ and $\alpha_{b,r} := c_{1,b}[r]$. This scalar is the hidden activation before the fixed $W_{2,b}$ synthesis. Let D_b denote the fixed LayerScale when present and the identity otherwise. We write the effective synthesis matrix as $V_b := D_b W_{2,b}$ and its rth write-back direction as $v_{b,r} := V_b[:, r]$. Throughout the paper, a *neuron* is an MLP hidden unit, $f_{b,r}$ is its *native function* on a token state, its value at one patch is a *token response*, and $\phi_{b,r}(x)$ is the corresponding attention-pooled *image coordinate*. A registered image coordinate is a *candidate atom*.

Global dictionary coordinates. Let $c_b(x)$ be the CLS state after block b, and let $\ell_b(x)$ be the result of applying the checkpoint-defined readout to $c_b(x)$, including its normalization and any existing output projection. Define $h_b(x) := \ell_b(x)/\max\{\|\ell_b(x)\|_2, \varepsilon_h\}$. The global companion is

$$h(x) := \frac{1}{\sqrt{L_{\mathcal{B}}}} \mathrm{Concat}_{b \in \mathcal{B}}\, h_b(x) \in \mathbb{R}^{L_{\mathcal{B}} d_h}, \qquad \|h(x)\|_2 \leq 1, \tag{3}$$

where d_h is the readout width and $\varepsilon_h > 0$ is a normalization floor. The normalization gives each depth a comparable scale, while the outer factor keeps the global-branch energy stable as $L_{\mathcal{B}}$ changes. Retaining several depths exposes global summaries produced at different stages of the frozen computation.

Attention-conditioned local coordinates. Let $A_{b,0p}^{(k)}(x)$ be the post-softmax attention from the CLS query to patch p in head k. All sums below exclude CLS and other special tokens. With H_a heads and nonzero patch-attention mass, define

$$a_{b,p}(x) := \frac{\sum_{k=1}^{H_a} A_{b,0p}^{(k)}(x)}{\sum_{p'=1}^{M} \sum_{k=1}^{H_a} A_{b,0p'}^{(k)}(x)}.$$

We use $a_{b,p}(x) = 1/M$ only in the zero-mass fallback. Thus $a_{b,p}(x) \geq 0$ and $\sum_p a_{b,p}(x) = 1$. Each native neuron induces the image-level function

$$\boxed{\phi_{b,r}(x) = \sum_{p=1}^{M} a_{b,p}(x)\sigma_b(w_{b,r}^\top u_{b,p}(x) + \alpha_{b,r}).} \tag{4}$$

Each coordinate combines what a neuron responds to with where the CLS query places attention. The token states have already incorporated interactions with other patches, so their local responses are evaluated in image context. Within a block, all neurons share the same spatial weights while retaining

their distinct pretrained response functions. Category differences concentrated in an object part, for example, can be exposed through these internal responses alongside the global summary $h(x)$.

Within each block, the pooled neuron responses form a kernel mean embedding of the attention-weighted patch states. The pretrained neuron functions determine the corresponding kernel (Appendix A.1).

Architectural Accessibility of Synthesis-Null Variation. Fixed post-CLS function banks can only transform the downstream global representation. The following result identifies hidden-response variation that native coordinates retain before checkpoint synthesis removes it.

Theorem 1 (Synthesis-null accessibility under hidden-response intervention)**.** *Fix the upstream state of block b. Let $z_{b,p} := [f_{b,r}(u_{b,p})]_{r=1}^{m_b} \in \mathbb{R}^{m_b}$ collect the hidden neuron responses at patch p, and write $z = \{z_{b,p}\}_p$. Let $H_0(z)$ be the resulting downstream global representation, and set $z_b^{\mathrm{att}}(z) := \sum_p a_{b,p} z_{b,p}$. For any fixed map r and perturbations $\delta = \{\delta_{b,p}\}_p$ with $\delta_{b,p} \in \ker(V_b)$, write $\Delta z_b^{\mathrm{att}} := z_b^{\mathrm{att}}(z+\delta) - z_b^{\mathrm{att}}(z)$.*

$$H_0(z+\delta) = H_0(z), \quad r(H_0(z+\delta)) = r(H_0(z)), \quad \Delta z_b^{\mathrm{att}} = \sum_p a_{b,p}\delta_{b,p}. \tag{5}$$

If $\Delta z_b^{\mathrm{att}} \neq 0$, the pooled native response distinguishes the intervention while every fixed post-CLS expansion remains unchanged.

Across the eight DINOv2 benchmarks, the synthesis-null component of the complete pre-registration native response contains 51.84–52.80% of its energy and accounts for 55.63–68.38% of its regularized objective decrease (Table 6). Appendix A.1 gives the architectural proof and derives the realized label-aligned criterion.

Checkpoint-only atom registration. Retaining all native atoms is often impractical, because Gram storage grows quadratically and the dense analytic solve grows cubically with the dictionary width. InnerCIL therefore registers a fixed budget of P atoms using checkpoint parameters alone. For a given frozen backbone, the resulting set $\mathcal{Q}_P$ is determined once and is independent of downstream data. For the affine LayerNorm parameters (γ_b, β_b), define an upper bound on the input norm and the registration score

$$R_b := \sqrt{d}\,\|\gamma_b\|_\infty + \|\beta_b\|_2, \qquad s_{b,r}^{\mathrm{reg}} := R_b \|w_{b,r}\|_2 \|v_{b,r}\|_2. \tag{6}$$

The P highest-scoring atoms form $\mathcal{Q}_P$. The factor $R_b\|w_{b,r}\|_2$ controls the variation of the neuron's pre-activation over the LayerNorm-reachable input set, while $\|v_{b,r}\|_2$ measures the magnitude with which that response is written back to the residual stream. Appendix A.1 derives its approximation guarantee. Figure 7 compares the omitted write-back variation under equal-width registration rules.

3.4 Fisher Refinement Analytic Classification

Checkpoint-only registration fixes P data-independent native atoms whose class-discriminative value depends on the observed data. Fisher Refinement Analytic Classification (FRAC) uses cumulative class moments to select discriminative native coordinates, then combines them with the global branch for analytic classification. Visual Fisher ranks their individual class separation, allocating a limited deployment budget and reducing the solve dimension.

Index the registered native functions by $q \in \{1, \ldots, P\}$ in a fixed order, with each q identifying one $(b, r) \in \mathcal{Q}_P$. The raw candidate feature is the column vector

$$\psi_P(x) := [h(x); \phi_1(x); \ldots; \phi_P(x); 1] \in \mathbb{R}^{D_P}, \qquad D_P := L_{\mathcal{B}} d_h + P + 1. \tag{7}$$

The candidate functions are fixed, whereas their deployed support and scaling remain stage dependent. With $N_{1:t}$ realized training views accumulated through task t, the raw-sum statistics are

$$G_t^{\mathrm{raw}} := \sum_{n=1}^{N_{1:t}} \psi_P(x_n)\psi_P(x_n)^\top, \qquad Q_t^{\mathrm{raw}} := \sum_{n=1}^{N_{1:t}} \psi_P(x_n)\mathbf{e}_{y_n}^\top. \tag{8}$$

For every seen class c and native coordinate q, retain the count n_c, mean $\mu_{t,cq}$, and population variance $\sigma_{t,cq}^2 := n_c^{-1} \sum_{n:y_n=c} (\phi_q(x_n) - \mu_{t,cq})^2$. We apply the classical Fisher separability criterion (Gu

et al., 2012) to these cumulative native-function statistics. Across the C_t seen classes, define $\overline{\mu}_{t,q} := {C_t}^{-1} \sum_c \mu_{t,cq}$. The Visual Fisher score and selected support are

$$B_{t,q} := {C_t}^{-1} \sum_c (\mu_{t,cq} - \overline{\mu}_{t,q})^2,$$
$$W_{t,q} := {C_t}^{-1} \sum_c \sigma^2_{t,cq}, \tag{9}$$
$$U_{t,q} := \frac{B_{t,q}}{\max(W_{t,q}, \varepsilon_{\mathrm{F}})}, \qquad J_t := \mathrm{TopK}_{q \le P}(U_{t,q}), \qquad \varepsilon_{\mathrm{F}} > 0.$$

Visual Fisher favors coordinates with strong class separation and low within-class variation. The selected coordinates join the global branch for ridge fitting, with the screening analysis given in Appendix A.2.

Coordinate amplitudes can vary substantially across the selected neurons. We use per-coordinate cumulative RMS scaling to remove this scale variation and balance the global and local branches without a dataset-specific fusion weight. Let $\rho_{t,q}$ be the cumulative RMS of coordinate q and set $c_{t,q} := (\sqrt{2K}\, \rho_{t,q})^{-1}$. With $D_K := L_{\mathcal{B}} d_h + K + 1$, the deployed feature is

$$\psi_t(x) := \left[\frac{h(x)}{\sqrt{2}}; \{c_{t,q} \phi_q(x)\}_{q \in J_t}; 1 \right]. \tag{10}$$

Let R_t denote the diagonal scaling in Equation (10). Appendix A.2 gives its exact form, the cumulative RMS definition, the statistic reparameterization, and the branch-balance derivation.

Analytic classification. At task t, the learner accumulates current-task raw sufficient statistics and class moments. It recomputes J_t, R_t, and Θ_t from the cumulative state. New classes add output columns to Q_t^{raw} and Θ_t. The selected and rescaled statistics (G_t, Q_t) are recovered exactly from $(G_t^{\mathrm{raw}}, Q_t^{\mathrm{raw}})$, and the analytic classifier is

$$\Theta_t := \arg\min_{\Theta} \sum_{n=1}^{N_{1:t}} \|\mathbf{e}_{y_n} - \Theta^\top \psi_t(x_n)\|_2^2 + \lambda \|\Theta\|_F^2 = (G_t + \lambda I)^{-1} Q_t. \tag{11}$$

Each observed training view has equal weight in this objective. The complete selected Gram matrix retains both global–local and local–local correlations, so the readout fits the retained coordinates jointly after class-balanced screening. The identity regularizer applies to every deployed coordinate, including the constant bias. For a test image, prediction is $\widehat{y}_t(x) := \arg\max_{c \in \mathcal{C}_{1:t}} [\psi_t(x)^\top \Theta_t]_c$.

3.5 Analytic Closure and Complexity

At every stage, InnerCIL exactly recovers the jointly trained optimum of its analytic model class: under the same deployed representation, it yields the unique ridge minimizer obtained from all classes and training samples observed so far. Sequential class arrival therefore introduces no additional optimization gap within this model class. Appendix A.3 gives the proof.

For a fixed candidate width, InnerCIL stores no state whose dimension grows with the number of previously observed images. Appendix A.5 details the persistent state and computational cost.

4 Experiments

4.1 Experimental Setup

Baselines. The CLIP (Radford et al., 2021) comparison includes ZS-CLIP (Radford et al., 2021), SimpleCIL (Zhou et al., 2025), RAPF (Huang et al., 2024), MG-CLIP (Huang et al., 2025), BOFA (Li et al., 2026), and AnaCP (Momeni et al., 2025). The DINOv2 (Oquab et al., 2023) comparison includes SLDA (Hayes & Kanan, 2020), GACL (Zhuang et al., 2024), MiN (Jiang et al., 2025), PrePrompt (Huang et al., 2026), RanPAC (McDonnell et al., 2023), LoRanPAC (Peng et al., 2024), LayUP (Ahrens et al., 2023), and AnaCP (Momeni et al., 2025).

Table 1: CLIP-based CIL results (%). Best results are red and second-best results are blue.

Method	CIFAR		Aircraft		CUB		Cars		Food		IN-R		UCF		Object		Average	
	A_{avg}	A_{last}	A_{avg}	A_{last}	A_{avg}	A_{last}	A_{avg}	A_{last}	A_{avg}	A_{last}	A_{avg}	A_{last}	A_{avg}	A_{last}	A_{avg}	A_{last}	A_{avg}	A_{last}
ZS-CLIP	81.81	71.38	26.66	17.22	74.38	63.06	82.60	76.37	87.86	81.92	83.37	77.17	75.50	67.64	38.43	26.43	68.83	60.15
SimpleCIL	84.15	76.63	59.24	48.09	83.81	77.52	92.04	86.85	87.89	81.65	81.06	74.48	90.44	85.68	52.06	40.13	78.84	71.38
RAPF	86.14	78.04	50.38	23.61	79.09	62.77	82.79	71.22	88.57	81.15	81.26	70.48	92.28	80.33	48.67	27.43	76.15	61.88
MG-CLIP	89.74	82.78	48.33	32.34	74.20	64.25	88.21	79.73	88.59	82.35	83.18	77.20	87.74	80.83	51.41	40.90	76.42	67.55
BOFA	86.41	79.48	69.94	59.67	87.03	80.75	93.00	90.45	89.04	83.05	85.39	79.73	92.02	88.08	59.47	47.16	82.79	76.05
AnaCP	88.34	81.37	71.86	62.80	88.80	83.41	92.62	87.59	95.13	92.06	88.52	83.11	92.06	87.57	73.39	62.60	86.34	80.06
AnaCP*	88.30	81.56	72.17	63.64	88.86	83.62	92.54	87.69	95.28	92.22	89.31	84.46	92.01	87.17	73.42	62.87	86.49	80.40
InnerCIL	90.53	85.26	77.55	70.63	88.93	83.93	94.04	90.52	95.49	92.70	89.61	84.81	92.52	89.15	75.53	66.05	88.03	82.88

Table 2: ViT-based CIL results (%). Best results are red and second-best results are blue.

Method	CIFAR		Aircraft		CUB		Cars		Food		IN-100		UCF		VTAB		Average	
	A_{avg}	A_{last}	A_{avg}	A_{last}	A_{avg}	A_{last}	A_{avg}	A_{last}	A_{avg}	A_{last}	A_{avg}	A_{last}	A_{avg}	A_{last}	A_{avg}	A_{last}	A_{avg}	A_{last}
SLDA	92.28	87.57	85.57	78.49	93.05	89.37	91.10	85.74	92.76	88.46	92.25	85.70	90.80	86.15	96.74	93.35	91.82	86.85
GACL	93.85	90.10	76.88	69.82	93.22	89.37	89.76	84.52	94.37	90.97	93.27	88.22	91.41	86.76	96.00	91.74	91.09	86.44
MiN	92.17	88.83	86.48	81.43	92.90	89.56	93.98	91.01	90.78	86.98	88.56	80.58	93.11	90.05	96.98	93.77	91.87	87.78
PrePrompt	93.02	88.11	71.47	55.45	91.58	87.73	85.26	80.49	90.19	85.14	86.34	77.37	89.46	84.59	93.85	88.87	87.65	80.97
RanPAC	94.00	90.23	83.69	76.54	93.47	90.25	90.76	87.13	95.43	92.32	92.70	88.06	90.73	87.70	96.73	93.50	92.19	88.22
LoRanPAC	93.98	89.90	83.51	76.57	93.60	90.42	91.02	87.61	95.32	92.13	93.04	88.20	90.72	87.72	96.60	92.13	92.22	88.09
LayUP	94.27	91.41	87.76	81.22	93.51	89.99	93.04	90.15	94.84	91.26	93.06	87.24	92.07	89.57	96.66	93.91	93.15	89.34
AnaCP	93.93	89.95	88.27	83.20	92.63	89.61	92.48	89.54	94.87	92.26	93.13	88.68	90.96	88.51	96.73	94.10	92.88	89.48
AnaCP*	95.50	92.21	88.80	83.77	93.67	90.10	93.13	90.65	94.91	92.21	93.30	88.70	92.10	89.79	96.75	94.17	93.52	90.20
InnerCIL	94.66	91.59	90.50	85.48	93.76	90.61	94.44	91.88	96.09	93.40	93.69	89.06	92.89	90.45	97.68	94.91	94.21	90.92

Datasets and protocols. Both comparisons use CIFAR-100 (Krizhevsky, 2009), FGVC-Aircraft (Maji et al., 2013), CUB-200-2011 (Wah et al., 2011), Cars-196 (Krause et al., 2013), Food-101 (Bossard et al., 2014), and UCF-101 (Soomro et al., 2012). The CLIP comparison additionally uses ImageNet-R (Hendrycks et al., 2021) and ObjectNet-200 (Barbu et al., 2019), whereas the ViT comparison uses ImageNet-100 (Deng et al., 2009) and VTAB-50 (Zhai et al., 2019). CIFAR-100, Aircraft, Food-101, ImageNet-100, and UCF-101 follow B0 Inc10. CUB-200, ImageNet-R, and ObjectNet-200 follow B0 Inc20, Cars-196 uses Inc10, and VTAB-50 follows B0 Inc5. For each dataset, all methods take the same splits, class orders, task schedules, and view augmentations. A_{avg} and A_{last} denote mean stage-wise and last-stage accuracy, respectively. Average is the mean over the datasets reported in each table.

Implementation details. The CLIP and ViT experiments use frozen CLIP ViT-B/16 and DINOv2 ViT-B/14, respectively. InnerCIL uses 224×224 inputs, one original and two augmented training views, and a single standard test view. We fix $P = 11264$, $K = 8704$, and $\lambda = 0.003$ across all datasets, both backbones, and all random seeds. Appendix A.4 gives the remaining settings and capacity-sensitivity results. Experiments run on a single NVIDIA GeForce RTX 4090 GPU.

AnaCP* denotes the complete published pipeline, and AnaCP omits Feature Generation to match our strict replay-free protocol.

4.2 Comparison with State-of-the-Art

Comparison with CLIP-based Methods. InnerCIL exceeds AnaCP* by $1.54/2.48$ points in Average $A_{\text{avg}}/A_{\text{last}}$ and ranks first on eight of eight datasets in A_{avg} and eight of eight in A_{last} (Table 1). Figure 3 reports task-wise gains on Cars, CIFAR, and ObjectNet.

Comparison with ViT-based Methods. InnerCIL exceeds AnaCP* by $0.69/0.72$ points in Average $A_{\text{avg}}/A_{\text{last}}$ and ranks first on six of eight datasets in A_{avg} and seven of eight in A_{last} (Table 2).

Results over three random seeds for both backbone configurations are reported in Appendix A.6.

4.3 Component Ablation

Table 3 builds InnerCIL from final-block CLS, adding multi-depth CLS and then the registered local dictionary with Random-K. Replacing Random-K with Visual Fisher isolates label-informed

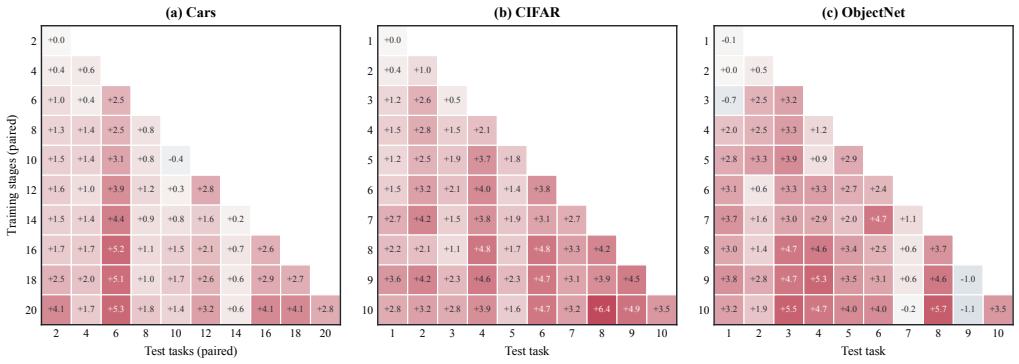


Figure 3: Task-wise CLIP accuracy gain of InnerCIL over AnaCP* on Cars, CIFAR, and ObjectNet. Cars pools adjacent training stages and test tasks in 2×2 blocks, weighted by evaluated sample counts. All panels use the same symmetric color scale.

Table 3: Progressive ablation on eight CLIP benchmarks (A_{last}, %). Δ_{Full} is the change in Average A_{last} relative to complete InnerCIL.

Configuration	CIFAR	Aircraft	CUB	Cars	Food	UCF	IN-R	Object	Average A_{last}	Δ_{Full}
Final-block CLS	76.99	55.09	75.89	79.52	89.15	83.28	70.57	53.91	73.05	−9.83
+ Multi-depth CLS	81.48	63.61	78.62	83.36	90.53	87.60	76.08	58.74	77.50	−5.38
+ Registered native dictionary (Random-K)	**85.40**	68.77	82.78	90.20	92.59	89.04	84.43	65.56	82.34	−0.54
+ Visual Fisher (InnerCIL)	85.26	**70.63**	**83.93**	**90.52**	**92.70**	**89.15**	**84.81**	**66.05**	**82.88**	**+0.00**

selection. All four stages share the data splits, class order, training views, and ridge readout across eight CLIP benchmarks.

Table 4 replaces individual components while fixing the global branch, P, K, ridge coefficient, and three training views. Post-CLS random ReLU tests the source of local capacity using random nonlinear functions of $h(x)$ (Appendix A.6). CLS-token native neurons evaluate the same pretrained functions at CLS, testing patch-level information. Uniform pooling averages the patch responses equally, testing attention's spatial weighting. Visible-only neurons remove the synthesis-null component, testing its contribution to classification (Appendix A.1).

Additional DINOv2 progressive and single-module ablations are reported in Appendix A.6.

Spatial contributions. Figure 4 localizes the target-logit contributions of the deployed native channels in the DINOv2 model. Appendix A.1 defines the decomposition.

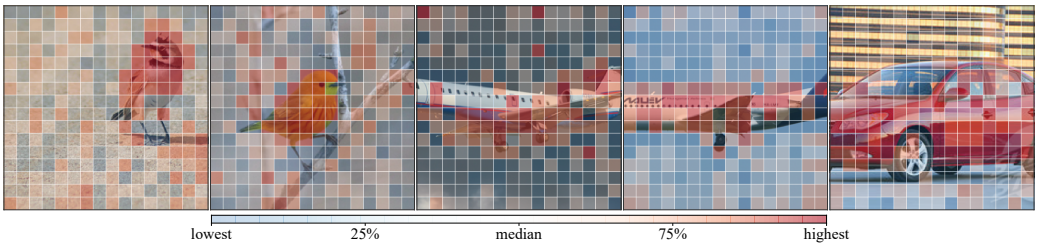


Figure 4: Target-logit contributions of the deployed native patch channels on CUB-200, FGVC-Aircraft, and Stanford Cars. Colors rank the 256 patches within each image from the lowest contribution (blue) to the highest contribution (red).

Synthesis-null contribution. Table 7 compares the unprojected native coordinates (All) with their synthesis-visible component (Vis). Before coordinate selection, the complete visible response is linearly equivalent to the pooled MLP write-back. Restoring the null component improves Average $A_{\text{avg}}/A_{\text{last}}$ by $0.81/1.39$ points with CLIP and $0.42/0.77$ with DINOv2, with positive A_{last} gains

Table 4: Equal-budget single-module replacements on eight CLIP benchmarks (A_{last}, %). Δ_{Full} is the change in Average A_{last} relative to complete InnerCIL. Best results are **bold**.

Single-module replacement	CIFAR	Aircraft	CUB	Cars	Food	UCF	IN-R	Object	Average A_{last}	Δ_{Full}
Post-CLS random ReLU	82.52	61.00	78.55	84.67	92.04	88.64	80.46	60.67	78.57	−4.31
CLS-token native neurons	83.92	64.45	78.51	83.50	91.79	87.84	79.28	61.56	78.86	−4.03
Uniform-pooled native neurons	**85.44**	64.39	72.11	88.24	92.56	86.96	83.98	62.24	79.49	−3.39
Visible-only native neurons	83.99	68.89	82.86	89.22	92.10	88.59	82.64	63.67	81.49	−1.39
InnerCIL	85.26	**70.63**	**83.93**	**90.52**	**92.70**	**89.15**	**84.81**	**66.05**	**82.88**	+0.00

in all 16 cases. This links pre-synthesis information to test performance at unchanged capacity. Appendix A.1 gives the decomposition, its proof, and the matched-classifier construction.

4.4 Computational and Storage Efficiency

Table 5 uses the same DINOv2 backbone, single GPU, and extraction batch size of 64 for both methods. **Feature** measures training-feature extraction, and **Method** measures classifier construction, including statistic accumulation and the analytic solve. **Time Total** sums them, excluding image loading, augmentation, and evaluation. **Memory** includes retained statistics, fitted parameters, and deployment metadata, excluding the frozen backbone and temporary workspace. Appendix A.5 details storage and complexity.

Table 5: Compute and persistent-state costs under a matched DINOv2 protocol. Each metric reports InnerCIL's cost and its signed difference from AnaCP* ($\Delta = \text{InnerCIL} - \text{AnaCP}^*$). Negative differences indicate lower costs, in seconds for time and MiB for storage.

Dataset	Feature (s)		Method (s)		Time Total (s)		Memory (MiB)	
	InnerCIL	Δ	InnerCIL	Δ	InnerCIL	Δ	InnerCIL	Δ
CIFAR	143.20	−50.91	15.21	−61.56	158.41	−112.47	415.36	−93.60
Aircraft	20.08	−6.10	11.97	−0.48	32.05	−6.58	415.36	−93.60
CUB	18.61	−4.92	12.36	+0.03	30.97	−4.89	438.41	−78.77
Cars	24.63	−7.13	23.21	+10.85	47.84	+3.72	437.49	−79.37
Food	218.27	−152.60	16.84	−96.13	235.10	−248.75	415.36	−93.60
IN-100	362.90	−138.00	21.13	−171.64	384.03	−309.64	415.36	−93.60
UCF	27.83	−9.45	12.04	−4.59	39.87	−14.03	415.36	−93.60
VTAB	6.00	−1.22	12.58	+8.16	18.58	+6.93	403.84	−101.01
Total	**821.52**	−370.35	**125.34**	−315.36	**946.86**	−685.71	**3356.55**	−727.16

InnerCIL has lower total compute time on six of eight datasets and lower persistent state on all eight, reducing their respective sums by 42.0% and 17.8%. Appendix A.5 reports InnerCIL's CLIP costs under the same measurement definitions.

5 Conclusion

Conclusions. We proposed InnerCIL, a novel analytic CIL framework built on pretrained internal neuron responses to achieve superior classifier construction. In InnerCIL, an ANDG mechanism constructs a native global–local dictionary in one frozen forward pass, and the FRAC scheme selects discriminative coordinates for closed-form classification. Theoretical analysis and experimental results show that InnerCIL preserves analytic closure under strict replay-free learning and achieves state-of-the-art performance with low computational and storage costs.

Limitations. InnerCIL requires internal activations, with Gram storage growing quadratically in dictionary width. A frozen backbone limits adaptation to domains poorly represented during pretraining.

Future work. We will compress second-order statistics and extend native dictionaries across architectures and modalities.

AI USE STATEMENT

Conceptual exploration, experimental design, code implementation, and the development of the research contributions were carried out entirely by the authors. Generative AI assisted only with manuscript polishing. The authors reviewed the AI-assisted work and take responsibility for the final content.

REPRODUCIBILITY STATEMENT

Section 3 describes the method, and Section 4 and Appendix A.4 provide the experimental protocols and implementation settings. The theoretical assumptions and proofs are given in Appendices A.1–A.3. We will publicly release the complete implementation and reproduction instructions upon acceptance.

REFERENCES

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. *arXiv preprint arXiv:2312.08888*, 2023.

Andrei Barbu, David Mayo, Julian Alverio, William Luo, Christopher Wang, Dan Gutfreund, Josh Tenenbaum, and Boris Katz. ObjectNet: A large-scale bias-controlled dataset for pushing the limits of object recognition models. *Advances in neural information processing systems*, 32, 2019.

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

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

Dipam Goswami, Yuyang Liu, Bartłomiej Twardowski, and Joost Van De Weijer. FeCAM: Exploiting the heterogeneity of class distributions in exemplar-free continual learning. *Advances in Neural Information Processing Systems*, 36:6582–6595, 2023.

Quanquan Gu, Zhenhui Li, and Jiawei Han. Generalized fisher score for feature selection. *arXiv preprint arXiv:1202.3725*, 2012.

Tyler L Hayes and Christopher Kanan. Lifelong machine learning with deep streaming linear discriminant analysis. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 887–896. IEEE, 2020.

Dan Hendrycks, Steven Basart, Norman Mu, Saurav Kadavath, Frank Wang, Evan Dorundo, Rahul Desai, Tyler Zhu, Samyak Parajuli, Mike Guo, et al. The many faces of robustness: A critical analysis of out-of-distribution generalization. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8320–8329. IEEE, 2021.

Libo Huang, Xiangqi Li, Jiarui Zhao, Zhulin An, Chuanguang Yang, Boyu Diao, Fei Wang, Yan Zeng, Zhifeng Hao, and Yongjun Xu. Preprompt: Predictive prompting for class incremental learning. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 1886–1897, 2026.

Linlan Huang, Xusheng Cao, Haori Lu, and Xialei Liu. Class-incremental learning with CLIP: Adaptive representation adjustment and parameter fusion. In *European Conference on Computer Vision*, pp. 214–231. Springer, 2024.

Linlan Huang, Xusheng Cao, Haori Lu, Yifan Meng, Fei Yang, and Xialei Liu. Mind the gap: Preserving and compensating for the modality gap in CLIP-based continual learning. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 3777–3786. IEEE, 2025.

Kai Jiang, Zhengyan Shi, Dell Zhang, Hongyuan Zhang, and Xuelong Li. Mixture of noise for pre-trained model-based class-incremental learning. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 44776–44802. Curran Associates, Inc., 2025. doi: 10.52202/085713-1491. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/3f54333743f508c891eaa4c8734d5235-Paper-Conference.pdf.

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*, pp. 554–561, 2013.

Alex Krizhevsky. Learning multiple layers of features from tiny images. Technical report, University of Toronto, 2009.

Lan Li, Tao Hu, Da-Wei Zhou, Jia-Qi Yang, Han-Jia Ye, and De-Chuan Zhan. BOFA: Bridge-layer orthogonal low-rank fusion for CLIP-based class-incremental learning. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(27):22967–22975, 2026.

Yan-Shuo Liang and Wu-Jun Li. InfLoRA: Interference-free low-rank adaptation for continual learning. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23638–23647. IEEE, 2024.

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

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.

Thomas Mensink, Jakob Verbeek, Florent Perronnin, and Gabriela Csurka. Distance-based image classification: Generalizing to new classes at near-zero cost. *IEEE transactions on pattern analysis and machine intelligence*, 35(11):2624–2637, 2013.

Saleh Momeni, Changnan Xiao, and Bing Liu. Anacp: Toward upper-bound continual learning via analytic contrastive projection. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 72923–72944. Curran Associates, Inc., 2025. doi: 10.52202/085713-2444. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/699b19e638a086ed3a6d1710c5aea504-Paper-Conference.pdf.

Maxime Oquab, Timothée Darcet, Théo Moutakanni, Huy Vo, Marc Szafraniec, Vasil Khalidov, Pierre Fernandez, Daniel Haziza, Francisco Massa, Alaaeldin El-Nouby, et al. DINOv2: Learning robust visual features without supervision. *arXiv preprint arXiv:2304.07193*, 2023.

Liangzu Peng, Juan Elenter, Joshua Agterberg, Alejandro Ribeiro, and René Vidal. LoRanPAC: Low-rank random features and pre-trained models for bridging theory and practice in continual learning. *arXiv preprint arXiv:2410.00645*, 2024.

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*, pp. 8748–8763. PmLR, 2021.

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 *2023 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 11909–11919. IEEE, 2023.

Alex Smola, Arthur Gretton, Le Song, and Bernhard Schölkopf. A hilbert space embedding for distributions. In *International conference on algorithmic learning theory*, pp. 13–31. Springer, 2007.

Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. UCF101: A dataset of 101 human actions classes from videos in the wild. *arXiv preprint arXiv:1212.0402*, 2012.

Catherine Wah, Steve Branson, Peter Welinder, Pietro Perona, and Serge Belongie. The Caltech-UCSD Birds-200-2011 dataset. 2011.

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

Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *2022 IEEE/CVF conference on computer vision and pattern recognition (CVPR)*, pp. 139–149. IEEE, 2022b.

Xiaohua Zhai, Joan Puigcerver, Alexander Kolesnikov, Pierre Ruyssen, Carlos Riquelme, Mario Lucic, Josip Djolonga, Andre Susano Pinto, Maxim Neumann, Alexey Dosovitskiy, et al. A large-scale study of representation learning with the visual task adaptation benchmark. *arXiv preprint arXiv:1910.04867*, 2019.

Gengwei Zhang, Liyuan Wang, Guoliang Kang, Ling Chen, and Yunchao Wei. SLCA: Slow learner with classifier alignment for continual learning on a pre-trained model. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 19091–19101. IEEE, 2023.

Da-Wei Zhou, Qi-Wei Wang, Zhi-Hong Qi, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. Class-incremental learning: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 46(12):9851–9873, 2024.

Da-Wei Zhou, Zi-Wen Cai, Han-Jia Ye, De-Chuan Zhan, and Ziwei Liu. Revisiting class-incremental learning with pre-trained models: Generalizability and adaptivity are all you need. *International Journal of Computer Vision*, 133(3):1012–1032, 2025.

Huiping Zhuang, Zhenyu Weng, Hongxin Wei, Renchunzi Xie, Kar-Ann Toh, and Zhiping Lin. ACIL: Analytic class-incremental learning with absolute memorization and privacy protection. *Advances in Neural Information Processing Systems*, 35:11602–11614, 2022.

Huiping Zhuang, Yizhu Chen, Di Fang, Run He, Kai Tong, Hongxin Wei, Ziqian Zeng, and Cen Chen. GACL: Exemplar-free generalized analytic continual learning. *Advances in neural information processing systems*, 37:83024–83047, 2024.

A APPENDIX

Contents

A.1 Native Dictionary Analysis

A.2 Visual Fisher and Deployment Geometry

A.3 Incremental Reconstruction and Algorithm

A.4 Implementation Settings and Capacity Sensitivity

A.5 Persistent State and Resource Measurements

A.6 Additional Ablations and Robustness

A.1 Native Dictionary Analysis

Proof of Theorem 1. Within block b, the MLP hidden response $z_{b,p}$ enters the residual stream only through $V_b z_{b,p}$. For $\delta_{b,p} \in \ker(V_b)$,

$$V_b(z_{b,p} + \delta_{b,p}) = V_b z_{b,p}. \tag{12}$$

The post-MLP token states are therefore unchanged, and the deterministic downstream blocks produce the same H_0. Any fixed composition $r(H_0)$ is unchanged as well. The CLS-to-patch attention weights used by InnerCIL are computed before this MLP hidden response. Holding the upstream state fixed therefore also fixes $a_{b,p}$, while linearity of attention pooling gives

$$\sum_p a_{b,p}(z_{b,p} + \delta_{b,p}) - \sum_p a_{b,p} z_{b,p} = \sum_p a_{b,p}\delta_{b,p}. \tag{13}$$

This proves Equation (5). The ambient annihilated space satisfies $\dim\ker(V_b) = m_b - \operatorname{rank}(V_b) \geq m_b - d$ by rank–nullity. Natural images realize a constrained subset of hidden responses. The label-aligned value of its realized synthesis-null component is characterized by Δ_t^{null} below.

Label-aligned synthesis-null variation. The following realized-sample criterion connects Theorem 1 to the analytic objective. Let N_b be the orthogonal projector onto $\ker(V_b)$. The complete pooled hidden vector decomposes into $z_b^{\text{vis}} = (I - N_b)z_b^{\text{att}}$ and $z_b^{\text{null}} = N_b z_b^{\text{att}}$. These are its orthogonal projections onto the row space and null space of V_b, respectively. Because attention pooling is linear and V_b is fixed, the pooled MLP write-back without its fixed output bias is

$$g_b(x) = \sum_p a_{b,p}(x) D_b\big(\mathrm{MLP}_b(u_{b,p}(x)) - c_{2,b}\big) = V_b z_b^{\text{att}}(x).$$

Since $V_b N_b = 0$,

$$V_b z_b^{\text{vis}} = V_b(I - N_b)z_b^{\text{att}} = g_b, \qquad V_b z_b^{\text{null}} = 0.$$

The row space of V_b intersects its kernel only at zero, so V_b is one-to-one on visible responses. Consequently, the complete visible response and pooled synthesis output are linearly interconvertible. A restriction to selected visible coordinates is a linear readout of this output. Because a general change of coordinates changes an isotropic ridge penalty, the matched classifier comparison below uses neuron coordinates throughout.

For a fixed audit sample set at stage t, let Z_t contain the complete native response rows and let Z_t^{vis} and Z_t^{null} contain their visible and null projections. Let H_t collect the global and synthesis-visible coordinates, and let $\mathbf{Y}$ collect the aligned one-hot targets for this sample set. Define

$$\begin{aligned} \Pi_{H,t}^{\lambda} &:= I - H_t(H_t^\top H_t + \lambda I)^{-1} H_t^\top, \\ A_t := (Z_t^{\text{null}})^\top \Pi_{H,t}^{\lambda} Z_t^{\text{null}} + \lambda I, &\qquad F_t := (Z_t^{\text{null}})^\top \Pi_{H,t}^{\lambda} \mathbf{Y}. \end{aligned} \tag{14}$$

The realized label-aligned synthesis-null value is

$$\Delta_t^{\text{null}} = \operatorname{tr}\big(F_t^\top A_t^{-1} F_t\big), \qquad \Delta_t^{\text{null}} > 0 \iff F_t \neq 0. \tag{15}$$

To see its operational meaning, let $\mathcal{L}^\star_{0,t}$ be the minimum regularized squared loss using H_t, and let $\mathcal{L}^\star_{1,t}$ be the minimum after adding Z_t^{null}. The joint normal equations are

$$\begin{bmatrix} H_t^\top H_t + \lambda I & H_t^\top Z_t^{\mathrm{null}} \\ (Z_t^{\mathrm{null}})^\top H_t & (Z_t^{\mathrm{null}})^\top Z_t^{\mathrm{null}} + \lambda I \end{bmatrix} \begin{bmatrix} \Theta_0 \\ \Theta_{\mathrm{null}} \end{bmatrix} = \begin{bmatrix} H_t^\top \mathbf{Y} \\ (Z_t^{\mathrm{null}})^\top \mathbf{Y} \end{bmatrix}.$$

Eliminating Θ_0 yields the Schur system $A_t \Theta_{\mathrm{null}} = F_t$, and completing the resulting quadratic gives

$$\mathcal{L}^\star_{0,t} - \mathcal{L}^\star_{1,t} = \Delta_t^{\mathrm{null}}. \tag{16}$$

Thus the synthesis-null channel has positive realized analytic value exactly when its ridge-residualized cross-statistic with the class targets is nonzero.

The split and complete native branches also attain the same minimum ridge objective. With a common scale for each block's visible and null responses, orthogonality gives

$$Z_t^{\mathrm{vis}} (Z_t^{\mathrm{vis}})^\top + Z_t^{\mathrm{null}} (Z_t^{\mathrm{null}})^\top = Z_t Z_t^\top.$$

Thus $[Z_t^{\mathrm{vis}}, Z_t^{\mathrm{null}}]$ and Z_t have identical sample kernels. Holding the global branch, targets, and isotropic ridge coefficient fixed preserves their minimum objectives, which depend on the feature matrix through this kernel.

Realized training-objective synthesis-null audit. We evaluate this criterion on frozen DINOv2 training representations before atom registration. For each dataset, we deterministically sample at most 25 original-view training images per class and decompose the complete 12,288-dimensional native response into synthesis-visible and synthesis-null components. A common native-branch scale preserves their orthogonal decomposition. Let Δ^{vis} denote the objective decrease from adding the visible component to the global representation, and let $\Delta^{\mathrm{all}} := \Delta^{\mathrm{vis}} + \Delta^{\mathrm{null}}$ be the decrease from adding the complete native response. We also report the mean coordinate leverage $(N_b)_{rr}$ of checkpoint-registered and equally sized fixed-random supports.

Table 6: Realized synthesis-null value in the complete pre-registration native response on training representations. Energy is the synthesis-null share of complete native-response energy. Null share is $\Delta^{\mathrm{null}}/\Delta^{\mathrm{all}}$. Leverage reports checkpoint/random registration.

Dataset	Null energy (%)	Δ^{vis}	Δ^{null}	Δ^{all}	Null share (%)	Leverage
CIFAR	51.84	48.11	63.44	111.55	56.87	0.746/0.750
Aircraft	52.80	48.01	103.81	151.81	68.38	0.746/0.750
CUB	51.90	86.43	177.70	264.13	67.28	0.746/0.750
Cars	52.47	197.82	352.81	550.62	64.07	0.746/0.750
Food	52.68	99.35	129.98	229.33	56.68	0.746/0.750
IN-100	52.11	19.39	37.04	56.43	65.64	0.746/0.750
UCF	52.14	52.01	65.19	117.20	55.63	0.746/0.750
VTAB	52.35	2.12	4.31	6.43	67.05	0.746/0.750

Synthesis-component classifiers. For each block, the extractor first pools all m_b hidden activations into z_b^{att}. An orthonormal basis Q_b for the row space of V_b, computed from its singular value decomposition, implements $z_b^{\mathrm{vis}} = Q_b(Q_b^\top z_b^{\mathrm{att}})$, with $N_b = I - Q_b Q_b^\top$. The projection therefore uses the complete synthesis matrix before the result is restricted to registered or deployed coordinates. Vis denotes the Visible-only control in Tables 4 and 11.

All computes J_t and $c_{t,q}$ from the unprojected registered dictionary using Visual Fisher and raw RMS. Vis and Null reuse these exact indices and scales at every stage, without reselecting or renormalizing either component. Let z_q^{vis} and z_q^{null} denote the corresponding scalar components at registered coordinate q. The deployed local vectors are

$$d_t^{\mathrm{vis}}(x) = [c_{t,q} z_q^{\mathrm{vis}}(x)]_{q \in J_t}, \qquad d_t^{\mathrm{null}}(x) = [c_{t,q} z_q^{\mathrm{null}}(x)]_{q \in J_t},$$
$$d_t^{\mathrm{all}}(x) = d_t^{\mathrm{vis}}(x) + d_t^{\mathrm{null}}(x) = [c_{t,q} \phi_q(x)]_{q \in J_t}.$$

Each local vector has K entries and is combined with the global branch and a unit bias. Global is the global-only analytic reference. All arms use identical samples, views, and class order, and fit

their own one-hot ridge classifier with the same λ. All$-$Vis thus measures restoration of the selected null response at fixed support, scale, and classifier dimension. Null$-$Global reports the difference between the local-augmented and global-only configurations.

Table 7: Test accuracy differences between synthesis-component classifiers. Global uses only the global branch. Vis, Null, and All add the projected visible component, its null complement, and their sum, respectively, sharing the selected support, RMS scales, and ridge regularization. Each entry reports $\Delta A_{\text{avg}}/\Delta A_{\text{last}}$ in percentage points.

CLIP			DINOv2		
Dataset	**Null − Global**	**All − Vis**	**Dataset**	**Null − Global**	**All − Vis**
CIFAR	$+2.00/+3.76$	$+0.54/+1.27$	CIFAR	$+1.14/+2.21$	$+0.30/+0.93$
Aircraft	$+6.19/+6.96$	$+1.64/+1.74$	Aircraft	$+1.58/+3.15$	$+0.95/+1.71$
CUB	$+3.48/+5.18$	$+0.70/+1.07$	CUB	$+0.29/+0.29$	$+0.20/+0.38$
Cars	$+4.24/+7.25$	$+0.75/+1.31$	Cars	$+1.93/+2.81$	$+0.43/+0.53$
Food	$+1.24/+2.05$	$+0.39/+0.60$	Food	$+1.45/+2.42$	$+0.60/+0.93$
IN-R	$+5.16/+8.62$	$+1.16/+2.17$	IN-100	$+0.41/+1.14$	$+0.19/+0.84$
UCF	$+0.30/+1.84$	$-0.05/+0.56$	UCF	$+0.81/+1.57$	$+0.44/+0.75$
Object	$+4.37/+7.48$	$+1.35/+2.38$	VTAB	$+0.49/+0.83$	$+0.21/+0.12$
Average	$\mathbf{+3.37/+5.39}$	$\mathbf{+0.81/+1.39}$	**Average**	$\mathbf{+1.01/+1.80}$	$\mathbf{+0.42/+0.77}$

Across all eight datasets, the synthesis-null component contains 51.84–52.80% of native-response energy, strictly decreases the audited regularized training objective, and accounts for 55.63–68.38% of the complete native objective decrease. This establishes label-aligned synthesis-null capacity in the source representation before axis registration and Visual Fisher selection. The classifier-level ablation in Table 7 further shows that this component improves test accuracy under both frozen backbones. Checkpoint registration prioritizes native axes according to their checkpoint write-back variation scores.

Figure 5 visualizes the effect of the realized synthesis-null coordinates through nearest-neighbor retrieval.

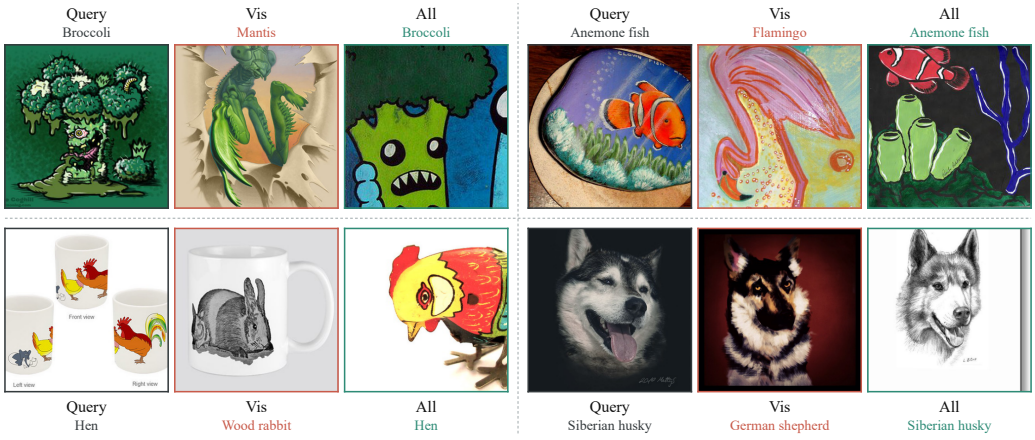
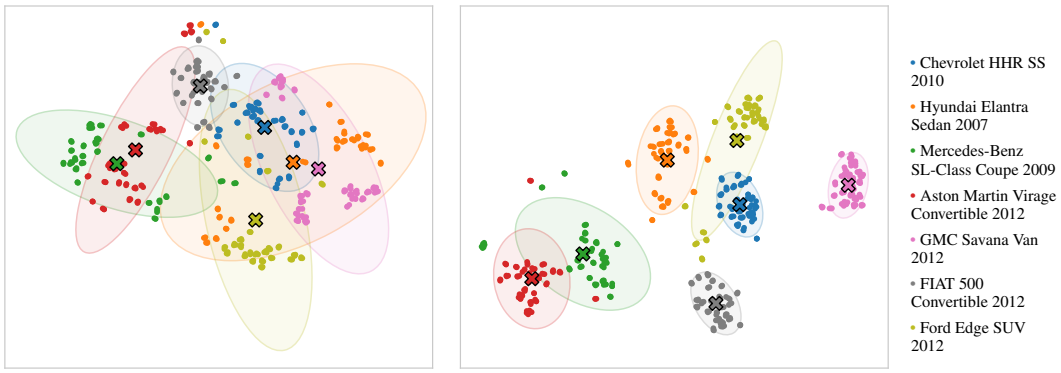


Figure 5: Nearest-neighbor corrections from realized synthesis-null coordinates on ImageNet-R. Each triplet shows the query image (left), its nearest neighbor using global and synthesis-visible coordinates (Vis, center), and its nearest neighbor using global and complete native coordinates (All, right). Red and green frames denote different- and same-class retrievals, respectively. The panels show four fixed queries from distinct classes.

Feature geometry. Figure 6 compares the native dictionary with an equal-budget post-CLS random ReLU dictionary. This CLIP visualization examines their local feature geometry on identical Cars examples.



Random post-CLS ReLU **InnerCIL native neurons**

Figure 6: Local feature geometry on seven selected classes from the first Stanford Cars task. The equal-budget random and native dictionaries use identical L2 normalization, PCA-50, and t-SNE settings.

Checkpoint registration guarantee. Write the affine LayerNorm entering block b's MLP as $u = \gamma_b \odot \widehat{s} + \beta_b$, where $\widehat{s} = (s - \bar{s}\mathbf{1})/\sqrt{d^{-1}\|s - \bar{s}\mathbf{1}\|_2^2 + \varepsilon_{\mathrm{LN}}}$. Then $\|\widehat{s}\|_2 \le \sqrt{d}$ and $\|u\|_2 \le R_b$, with R_b defined in Equation (6). For a support S, define the uncentered checkpoint write-back fields

$$g_b(x) := \sum_r v_{b,r}\phi_{b,r}(x), \qquad g_{b,S}(x) := \sum_{r:(b,r)\in S} v_{b,r}\phi_{b,r}(x). \tag{17}$$

Let $L_\sigma := \max_{b\in\mathcal{B}} \mathrm{Lip}(\sigma_b)$ denote a common Lipschitz constant for the fixed activations in these blocks.

Lemma 2 (Top-P control of checkpoint write-back variation)**.** *For every $|S| = P$, the fields in Equation* (17) *satisfy*

$$\sup_{x,x'} \left(\sum_{b\in\mathcal{B}} \|(g_b - g_{b,S})(x) - (g_b - g_{b,S})(x')\|_2^2 \right)^{1/2} \le 2L_\sigma \sum_{(b,r)\notin S} s_{b,r}^{\mathrm{reg}}. \tag{18}$$

Moreover, $\mathcal{Q}_P$ minimizes the right-hand side over all size-P supports.

The lemma characterizes registration as the minimizer of a separable worst-case envelope on omitted checkpoint write-back variation. The same control extends to bounded readouts aligned with the checkpoint synthesis geometry.

Corollary 3 (Approximation of checkpoint-aligned readouts)**.** *For matrices $U_b \in \mathbb{R}^{d\times C}$, define $F_U(x) := \sum_b g_b(x)^\top U_b$ and $F_{U,S}(x) := \sum_b g_{b,S}(x)^\top U_b$. If $\sum_b \|U_b\|_2^2 \le \Gamma^2$, then*

$$\sup_{x,x'} \|(F_U - F_{U,S})(x) - (F_U - F_{U,S})(x')\|_2 \le 2\Gamma L_\sigma \sum_{(b,r)\notin S} s_{b,r}^{\mathrm{reg}}. \tag{19}$$

Checkpoint write-back envelope proof. Let g_b be the uncentered field in Equation (17), and let $g_{b,S}$ retain only indices in S. Because the attention weights have unit mass, the constant $f_{b,r}(0)$ cancels between any two images. By Equation (6) and Lipschitz continuity,

$$|\phi_{b,r}(x) - \phi_{b,r}(x')| \le \sum_p a_{b,p}(x)|f_{b,r}(u_{b,p}(x)) - f_{b,r}(0)| \tag{20}$$

$$+ \sum_p a_{b,p}(x')|f_{b,r}(u_{b,p}(x')) - f_{b,r}(0)| \tag{21}$$

$$\le 2L_\sigma R_b \|w_{b,r}\|_2. \tag{22}$$

For $d_{b,S} := g_b - g_{b,S}$, this gives

$$\|d_{b,S}(x) - d_{b,S}(x')\|_2 \le 2L_\sigma \sum_{r:(b,r)\notin S} R_b \|w_{b,r}\|_2 \|v_{b,r}\|_2 = 2L_\sigma \sum_{r:(b,r)\notin S} s^{\mathrm{reg}}_{b,r}. \tag{23}$$

Using $\|(d_b)_b\|_2 \le \sum_b \|d_b\|_2$ gives Equation (18). If $s^{\mathrm{reg}}_i > s^{\mathrm{reg}}_j$ with $i \notin S$ and $j \in S$, exchanging j for i strictly decreases the omitted score sum. Repeated exchange terminates at the P largest scores, proving optimality for the stated separable envelope.

For Corollary 3, Cauchy–Schwarz across blocks gives

$$\left\| \sum_b [d_{b,S}(x) - d_{b,S}(x')]^\top U_b \right\|_2 \le \left(\sum_b \|d_{b,S}(x) - d_{b,S}(x')\|_2^2 \right)^{1/2} \left(\sum_b \|U_b\|_2^2 \right)^{1/2}. \tag{24}$$

Applying Lemma 2 proves Equation (19). Moreover, $g_{b,S}(x)^\top U_b = \sum_{r:(b,r)\in S} \phi_{b,r}(x) v_{b,r}^\top U_b$, so this function family is implemented by linear readouts of the retained scalar atoms.

Figure 7 evaluates the empirical pairwise counterpart of the write-back criterion. Each pair's omitted variation is normalized by its complete write-back variation:

$$\frac{\left(\sum_b \|d_{b,S}(x) - d_{b,S}(x')\|_2^2\right)^{1/2}}{\max\left\{\left(\sum_b \|g_b(x) - g_b(x')\|_2^2\right)^{1/2}, \varepsilon_{\mathrm{pair}}\right\}}, \qquad \varepsilon_{\mathrm{pair}} = 10^{-10}.$$

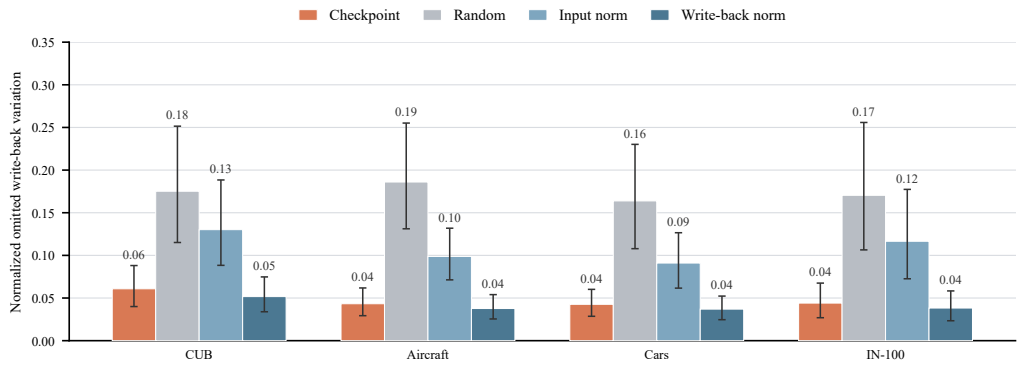


Figure 7: Normalized checkpoint write-back variation omitted by equal-width atom registration. Bars report the median over 4,096 deterministic training-image pairs, and whiskers show the interquartile range. Checkpoint registration omits less realized pairwise write-back variation than fixed random or input-norm registration on all four datasets. Direct write-back-norm ranking is included as a structural reference.

Kernel-mean interpretation. Define the input-conditioned token-state probability measure

$$\nu_{b,x} := \sum_{p=1}^{M} a_{b,p}(x) \delta_{u_{b,p}(x)}. \tag{25}$$

Then

$$\phi_{b,r}(x) = \int f_{b,r}(u)\, \mathrm{d}\nu_{b,x}(u).$$

Let $\mathcal{Q}_{P,b}$ be the subset of $\mathcal{Q}_P$ in block b, and define

$$\varphi_{b,P}(u) := [f_{b,r}(u)]_{(b,r)\in\mathcal{Q}_{P,b}},$$
$$k_{b,P}(u,u') := \langle \varphi_{b,P}(u), \varphi_{b,P}(u') \rangle.$$

The registered uncentered local vector before RMS scaling is therefore the finite kernel mean embedding (Smola et al., 2007)

$$\Phi_{b,P}(x) = \int \varphi_{b,P}(u)\,\mathrm{d}\nu_{b,x}(u). \tag{26}$$

The checkpoint fixes the finite kernel coordinates, while the input fixes the token-state measure embedded in those coordinates. Its induced image-level kernel is

$$\begin{aligned} K_{b,P}(x,x') &:= \langle \Phi_{b,P}(x), \Phi_{b,P}(x') \rangle \\ &= \sum_{p,p'} a_{b,p}(x) a_{b,p'}(x') k_{b,P}(u_{b,p}(x), u_{b,p'}(x')). \end{aligned} \tag{27}$$

Thus local image similarity is the attention-weighted average of all cross-image patch-state similarities measured by the checkpoint's registered neuron responses. After support selection and RMS deployment, the complete image kernel is

$$\kappa_t(x,x') = \frac{1}{2}\langle h(x), h(x') \rangle + \sum_{q \in J_t} c_{t,q}^2 \phi_q(x) \phi_q(x') + 1. \tag{28}$$

The analytic classifier is therefore ridge regression with a finite, stage-dependent kernel combining multi-depth global coordinates, selected native local coordinates, and the bias.

Patch-level target-logit contributions. Figure 4 uses the final-stage classifier and the ground-truth class y of each image. For a selected coordinate q associated with (b, r), let $\theta_{t,qy}$ denote its local readout coefficient in Θ_t. Its contribution at patch p is

$$d_{t,q,p}(x,y) := a_{b,p}(x) f_{b,r}(u_{b,p}(x)) c_{t,q} \theta_{t,qy}.$$

The displayed patch score is $\sum_{q \in J_t} d_{t,q,p}(x,y)$, aggregating selected channels across blocks at the same spatial position. Summing these scores over patches recovers the local branch's target logit, $\sum_{q \in J_t} c_{t,q} \phi_q(x) \theta_{t,qy}$, excluding the global branch and bias. Colors encode within-image ranks, from the lowest score in blue to the highest in red, rather than the signs of the contributions.

A.2 Visual Fisher and Deployment Geometry

Cumulative RMS deployment. For every native coordinate, the floor-stabilized cumulative RMS is

$$\rho_{t,q} := \sqrt{\max\left(\frac{1}{N_{1:t}} \sum_{n=1}^{N_{1:t}} \phi_q(x_n)^2, \varepsilon_{\mathrm{RMS}} \right)}, \qquad \varepsilon_{\mathrm{RMS}} > 0. \tag{29}$$

Let $E_t \in \mathbb{R}^{D_P \times D_K}$ select the global coordinates, J_t, and the bias. Let R_t apply the global scale $2^{-1/2}$, the local scales $c_{t,q} := (\sqrt{2K}\rho_{t,q})^{-1}$, and the unit bias scale. Then

$$\begin{aligned} T_t &:= E_t R_t, & \psi_t(x) &= T_t^\top \psi_P(x), \\ G_t &= T_t^\top G_t^{\mathrm{raw}} T_t, & Q_t &= T_t^\top Q_t^{\mathrm{raw}}. \end{aligned} \tag{30}$$

Thus every stage-dependent support and RMS scale is applied directly to the cumulative raw statistics without revisiting previous feature rows.

Branch energy control.

$$\frac{1}{N_{1:t}} \sum_n \left\| h(x_n)/\sqrt{2} \right\|_2^2 \le \frac{1}{2}, \qquad \frac{1}{N_{1:t}} \sum_n \sum_{q \in J_t} c_{t,q}^2 \phi_q(x_n)^2 \le \frac{1}{2}. \tag{31}$$

The first inequality follows from $\|h(x)\|_2 \le 1$, with equality when the global normalization floor is inactive. The local inequality follows by substitution of $c_{t,q}$ and becomes equality when no selected RMS floor is active.

RMS invariance and induced regularization. If a raw local coordinate is rescaled as $\phi'_q = a_q\phi_q$ with $a_q \neq 0$, and the RMS floor is inactive before and after rescaling, then $\rho'_q = |a_q|\rho_q$ and its deployed value changes only by $\mathrm{sign}(a_q)$. Hence positive changes of a neuron's amplitude unit leave the deployed coordinate invariant. More generally, if X is a raw feature matrix and D is its diagonal deployment scaling, then

$$\min_{\Theta} \|\mathbf{Y} - XD\Theta\|_F^2 + \lambda\|\Theta\|_F^2 = \min_{B} \|\mathbf{Y} - XB\|_F^2 + \lambda\|D^{-1}B\|_F^2. \tag{32}$$

RMS deployment therefore both normalizes coordinate energy and induces a second-moment-adaptive diagonal penalty in the raw parameterization.

Class-balanced screening risk. Let $W_{t,q}^+ := \max(W_{t,q}, \varepsilon_{\mathrm{F}})$. The bounded score associated with Visual Fisher is

$$\nu_{t,q} := \frac{U_{t,q}}{1 + U_{t,q}} = \frac{B_{t,q}}{B_{t,q} + W_{t,q}^+}. \tag{33}$$

The additive screening objective is

$$\mathcal{A}_t(S) := \sum_{q \in S} \nu_{t,q}. \tag{34}$$

Consider the class-balanced empirical distribution over the seen training views: classes have probability $1/C_t$ and samples are uniform within each class. Let $\Phi := [\phi_1, \ldots, \phi_P]^\top$, $Z := \Phi - \mathbb{E}\Phi$, and $\overline{\mathbf{e}}_Y := \mathbf{e}_Y - C_t^{-1}\mathbf{1}$. Define

$$\begin{aligned} \Sigma_t &:= \mathbb{E}[ZZ^\top], \qquad M_t := \mathbb{E}[Z\overline{\mathbf{e}}_Y^\top], \\ \widehat{\Sigma}_t &:= \Sigma_t + \mathrm{Diag}\Big(\big[W_{t,q}^+ - W_{t,q}\big]_{q=1}^P\Big), \\ D_t &:= \mathrm{Diag}\Big(\big[\widehat{\Sigma}_{t,qq}\big]_{q=1}^P\Big). \end{aligned} \tag{35}$$

For a support S with $\widehat{\Sigma}_{t,SS} \succ 0$, let $\Delta_t^{\mathrm{coord}}(S) := \mathrm{tr}(M_{t,S}^\top \widehat{\Sigma}_{t,SS}^{-1} M_{t,S})$ be the floor-regularized class-target energy captured jointly by coordinates S. Define the standardized covariance $\mathcal{R}_{t,S} := D_{t,S}^{-1/2}\widehat{\Sigma}_{t,SS}D_{t,S}^{-1/2}$.

Theorem 4 (Visual Fisher as additive one-coordinate risk reduction)**.** *For each coordinate q, class-balanced one-coordinate regression with the variance-floor penalty satisfies*

$$\min_{\theta \in \mathbb{R}^{C_t}} \big\{\mathbb{E}\|\overline{\mathbf{e}}_Y - Z_q\theta\|_2^2 + (W_{t,q}^+ - W_{t,q})\|\theta\|_2^2\big\} = 1 - \frac{1}{C_t} - \frac{\nu_{t,q}}{C_t}. \tag{36}$$

Hence Visual Fisher selects the K coordinates with the largest individual One-Hot risk reductions relative to the class-mean predictor, with $\Delta_t^{\mathrm{coord}}(\{q\}) = \nu_{t,q}/C_t$ and $J_t \in \arg\max_{|S|=K} \mathcal{A}_t(S)$.

For the joint-coordinate analysis, assume every K-coordinate principal submatrix of $\widehat{\Sigma}_t$ is positive definite. For $|S| = K$, set $L_{t,S} := D_{t,S}^{-1/2}M_{t,S}$. Then

$$\Delta_t^{\mathrm{coord}}(S) = \frac{\mathcal{A}_t(S)}{C_t} + \mathrm{tr}\Big[L_{t,S}^\top(\mathcal{R}_{t,S}^{-1} - I)L_{t,S}\Big]. \tag{37}$$

The displayed trace is the correlation contribution that separates this additive objective from the joint multivariate class-target energy. If $S_t^\star \in \arg\max_{|S|=K} \Delta_t^{\mathrm{coord}}(S)$, then

$$\Delta_t^{\mathrm{coord}}(J_t) \geq \frac{\lambda_{\min}(\mathcal{R}_{t,S_t^\star})}{\lambda_{\max}(\mathcal{R}_{t,J_t})}\Delta_t^{\mathrm{coord}}(S_t^\star). \tag{38}$$

In particular, Visual Fisher recovers an optimal support for $\Delta_t^{\mathrm{coord}}$ when the standardized coordinates are uncorrelated on every feasible support.

Proof. The qth row of M_t is $C_t^{-1}[\mu_{t,1q} - \overline{\mu}_{t,q}, \ldots, \mu_{t,C_tq} - \overline{\mu}_{t,q}]$, so

$$\|M_{t,q:}\|_2^2 = \frac{B_{t,q}}{C_t}, \qquad (D_t)_{qq} = B_{t,q} + W_{t,q}^+. \tag{39}$$

Since $\mathbb{E}\|\overline{\mathbf{e}}_Y\|_2^2 = 1 - 1/C_t$, the objective in Equation (36) expands to

$$1 - \frac{1}{C_t} - 2M_{t,q:}\theta + (B_{t,q} + W_{t,q}^+)\|\theta\|_2^2.$$

Its minimizer is $\theta = M_{t,q:}^\top/(B_{t,q} + W_{t,q}^+)$. Substitution and Equation (39) give the one-coordinate risk identity and $\Delta_t^{\mathrm{coord}}(\{q\}) = \nu_{t,q}/C_t$. Summing the independently minimized risks over a size-K support gives $K(1 - 1/C_t) - \mathcal{A}_t(S)/C_t$. Since $u \mapsto u/(1+u)$ is strictly increasing, Top-K by $U_{t,q}$ minimizes this sum.

For any S,

$$\Delta_t^{\mathrm{coord}}(S) = \mathrm{tr}\Big(L_{t,S}^\top \mathcal{R}_{t,S}^{-1} L_{t,S}\Big), \qquad \|L_{t,S}\|_F^2 = \frac{\mathcal{A}_t(S)}{C_t}. \tag{40}$$

Adding and subtracting $\|L_{t,S}\|_F^2$ proves the first identity in Equation (37). The extremal eigenvalues of $\mathcal{R}_{t,S}$ also give the support-specific bounds

$$\frac{\mathcal{A}_t(S)}{C_t\lambda_{\max}(\mathcal{R}_{t,S})} \le \Delta_t^{\mathrm{coord}}(S) \le \frac{\mathcal{A}_t(S)}{C_t\lambda_{\min}(\mathcal{R}_{t,S})}. \tag{41}$$

Applying the lower bound to J_t, using $\mathcal{A}_t(J_t) \ge \mathcal{A}_t(S_t^\star)$, and applying the upper bound to $S_t^\star$ proves Equation (38). When $\mathcal{R}_{t,S} = I$ for every feasible support, the correlation correction vanishes and the result is exact.

With inactive variance floors, $\widehat{\Sigma}_t = \Sigma_t$ and $\mathrm{rank}(\Sigma_t) \le N_{1:t} - 1$. The positive-definiteness assumption for the joint-coordinate analysis therefore requires $K < N_{1:t}$, as well as full rank on each feasible support. The singleton risk identity and additive Top-K optimum hold independently of this joint rank condition.

Screening and the deployed ridge objective. The one-coordinate risks above give equal probability to each class and use centered responses to measure class separation. Their variance-floor penalty is exactly $W_{t,q}^+ - W_{t,q}$, which vanishes when the floor is inactive. The classifier instead uses the uncentered, RMS-scaled coordinates in Equation (10), including the global branch and the regularized bias, and gives equal weight to each training view. For a size-K support S, let $G_t(S)$ and $Q_t(S)$ denote its complete deployed Gram and label cross-statistic. Substituting the ridge solution into Equation (11) gives the minimum objective

$$\|\mathbf{Y}_{1:t}\|_F^2 - \mathrm{tr}\big[Q_t(S)^\top (G_t(S) + \lambda I)^{-1} Q_t(S)\big]. \tag{42}$$

This joint fit uses class frequencies, cumulative RMS scales, and all cross-statistics among retained global and local coordinates. Visual Fisher chooses the support by minimizing the sum of the one-coordinate risks in Equation (36). At that support, ridge fitting minimizes Equation (11) over classifier weights, attaining the value in Equation (42). The correlation correction and approximation bound in Theorem 4 apply to the class-balanced coordinate objective $\Delta_t^{\mathrm{coord}}$.

Class-moment updates and support reassessment. The Visual Fisher statistics can be merged exactly without retaining feature rows. For one class and one native coordinate, the raw count, sum, and square sum determine the count, mean, and centered sum of squares (n, μ, M_2). Let (m, ν, S_2) denote the corresponding statistics of a new batch. Their union has

$$\begin{aligned} n' &= n + m, \\ \mu' &= \frac{n\mu + m\nu}{n'}, \\ M_2' &= M_2 + S_2 + \frac{nm}{n'}(\mu - \nu)^2, \qquad (\sigma')^2 = \frac{M_2'}{n'}. \end{aligned} \tag{43}$$

The equations apply coordinate-wise to all registered atoms and are associative, so task-wise and batch aggregation produce identical class moments. The implementation adds raw counts, sums, and square sums, from which these means and variances are recovered.

The between-class term also admits an exact class-arrival decomposition. Let $\mathcal{C}_O$ and $\mathcal{C}_N$ be disjoint old and new class sets, with C_O and C_N classes. For a fixed coordinate q, let $B_{O,q}$ and $B_{N,q}$ be their population variances of class means, and let $\overline{\mu}_{O,q}$ and $\overline{\mu}_{N,q}$ be their group means. With $\alpha = C_O/(C_O + C_N)$,

$$B_{O\cup N,q} = \alpha B_{O,q} + (1-\alpha)B_{N,q} + \alpha(1-\alpha)(\overline{\mu}_{O,q} - \overline{\mu}_{N,q})^2. \tag{44}$$

The final term measures separation between the arriving and previously seen class groups. Consequently, a fixed registered dictionary can acquire a different Fisher ordering as classes arrive, while all quantities required to recompute that ordering remain class-level aggregates.

A.3 Incremental Reconstruction and Algorithm

Sufficient-statistic batch equivalence. Assume exact arithmetic, identical realized features, and aligned class columns. The raw Gram and label cross-statistic are additive. Let $\mathcal{S}_t^{\mathrm{cls}}$ collect the per-class counts, coordinate sums, and coordinate square sums. The counts, means, and centered sums of squares maintained by Equation (43) are an equivalent representation of these raw class moments. If ΔG_τ, ΔQ_τ, and $\Delta\mathcal{S}_\tau^{\mathrm{cls}}$ denote the task-τ increments of the Gram, label cross-statistic, and class-moment state, then both incremental and batch aggregation produce

$$G_t^{\mathrm{raw}} = \sum_{\tau=1}^{t} \Delta G_\tau, \qquad Q_t^{\mathrm{raw}} = \sum_{\tau=1}^{t} \Delta Q_\tau, \qquad \mathcal{S}_t^{\mathrm{cls}} = \sum_{\tau=1}^{t} \Delta\mathcal{S}_\tau^{\mathrm{cls}}. \tag{45}$$

The support map and diagonal deployment map are deterministic functions of this common state. Both procedures obtain the same J_t and R_t, and hence identical normal equations. Because $\lambda > 0$, $G_t + \lambda I$ is positive definite. Its solution Θ_t is unique, establishing equality between the incremental classifier and the corresponding batch ridge classifier at stage t.

Incremental algorithm. Algorithm 1 summarizes the complete incremental procedure. Checkpoint registration is performed once. Each task contributes only raw outer products, label cross-products, coordinate square sums, and class moments. Support selection, deployment scaling, and the classifier are reconstructed from their cumulative values.

Algorithm 1 InnerCIL training and prediction

Require: Class-incremental stream of labeled images
1: $\mathcal{Q}_P \leftarrow$ CheckpointRegister
2: Initialize $G^{\mathrm{raw}} \leftarrow 0$, $Q^{\mathrm{raw}} \leftarrow 0$, and per-class native-coordinate moments
3: **for** $t = 1, \ldots, T$ **do**
4: Append zero columns to Q^{raw} for classes in $\mathcal{C}_t$
5: **for** each realized training view (x, y) from $\mathcal{D}_t$ **do**
6: Run the frozen ViT once to obtain $h(x)$, $u_{b,p}(x)$, and $a_{b,p}(x)$
7: Evaluate $\phi_1(x), \ldots, \phi_P(x)$ and assemble $\psi_P(x)$
8: $G^{\mathrm{raw}} \leftarrow G^{\mathrm{raw}} + \psi_P(x)\psi_P(x)^\top$
9: $Q^{\mathrm{raw}} \leftarrow Q^{\mathrm{raw}} + \psi_P(x)\mathbf{e}_y^\top$
10: Update the moments of class y
11: **end for**
12: Compute $U_{t,q}$ from cumulative class moments and set $J_t \leftarrow \mathrm{TopK}_{q\le P}(U_{t,q})$
13: Recover cumulative RMS values $\rho_{t,q}$ and construct $T_t = E_t R_t$
14: $G_t \leftarrow T_t^\top G^{\mathrm{raw}} T_t$, $Q_t \leftarrow T_t^\top Q^{\mathrm{raw}}$
15: $\Theta_t \leftarrow$ AnalyticSolve(G_t, Q_t)
16: **end for**
17: **return** $\widehat{y}_t(x) = \arg\max_{c\in\mathcal{C}_{1:t}} [\psi_t(x)^\top \Theta_t]_c$

A.4 Implementation Settings and Capacity Sensitivity

Dataset subsets and task partitions. Food-101 and UCF-101 are evaluated as 100-class subsets. For each dataset, one class is selected at random and excluded from both the training and test splits.

This choice is made once, and the retained class subset is shared by all methods, both backbones, and all three random seeds. The benchmark train/test assignments of the retained images are preserved.

B0 denotes a sequence without a separate base session, and Inc$q$ denotes the number of classes introduced per task. CIFAR-100, Aircraft, Food, ImageNet-100, and UCF each contain ten tasks of ten classes. CUB-200, ImageNet-R, and ObjectNet-200 each contain ten tasks of twenty classes. Cars-196 contains nineteen tasks of ten classes followed by one task of six classes, retaining all 196 classes. VTAB-50 contains ten tasks of five classes. For each dataset and seed, the same class order determines the task partitions for every compared method. Each stage evaluates a single classifier on the test images of all seen classes, without access to task identity.

Representation and training views. The CLIP and DINOv2 experiments use $\mathcal{B} = \{8, 9, 10, 11\}$. These late blocks combine repeated patch-token interactions with spatially resolved intermediate states. All benchmarks use the same dictionary width $P = 11264$, deployment width $K = 8704$, and ridge coefficient $\lambda = 0.003$.

The seeds 1993, 2024, and 3407 determine class order and method-specific randomness, with the training views held fixed across runs. Each training image provides one standard view and two independently augmented 224×224 views, using random resized cropping with area scale $[0.65, 1.0]$, horizontal flipping with probability 0.5, and checkpoint-specific normalization. Evaluation uses one standard view per test image.

All compared methods are reproduced locally under the same dataset subsets, train/test splits, class orders, task partitions, and view-augmentation protocol. The ablation controls retain these conditions, so comparisons isolate the method or component being evaluated.

Capacity sensitivity. Table 8 and Figure 8 compare dictionary and deployment widths with the remaining settings held fixed. Each configuration is applied to all eight datasets under each backbone.

Table 8: Capacity sensitivity at $\lambda = 0.003$. CLIP and DINOv2 columns report Average A_{last} (%) over their eight benchmarks, and Joint averages all 16 backbone–dataset cases.

P	K	CLIP	DINOv2	Joint	P	K	CLIP	DINOv2	Joint
8192	4096	81.82	90.53	86.17	11264	7424	82.79	90.93	86.86
8192	6144	82.04	90.79	86.41	11264	8192	82.86	90.92	86.89
9216	4096	82.08	90.54	86.31	11264	8704	82.88	90.92	86.90
9216	6144	82.36	90.85	86.61	11264	9216	82.91	90.87	86.89
9216	7424	82.44	90.86	86.65	11264	9728	82.93	90.82	86.87
10240	4096	82.32	90.54	86.43	11520	9216	82.90	90.88	86.89
10240	6144	82.60	90.91	86.76	11776	4096	82.41	90.55	86.48
10240	6912	82.62	90.92	86.77	11776	6144	82.71	90.89	86.80
10240	7168	82.70	90.92	86.81	11776	7424	82.73	90.93	86.83
10240	7424	82.71	90.95	86.83	11776	8192	82.77	90.92	86.85
10240	7680	82.70	90.93	86.82	11776	8704	82.85	90.91	86.88
10240	7936	82.76	90.91	86.83	11776	8960	82.93	90.90	86.92
10240	8192	82.76	90.94	86.85	11776	9216	82.96	90.85	86.91
10240	8704	82.76	90.87	86.81	11776	9472	82.93	90.84	86.89
10752	8704	82.76	90.87	86.82	11776	9728	82.92	90.83	86.88
11264	4096	82.41	90.62	86.51	11776	10240	82.91	90.84	86.88
11264	6144	82.61	90.91	86.76					

A.5 Persistent State and Resource Measurements

InnerCIL resource costs with CLIP. Table 9 reports CLIP costs using the definitions and exclusions in Section 4.4.

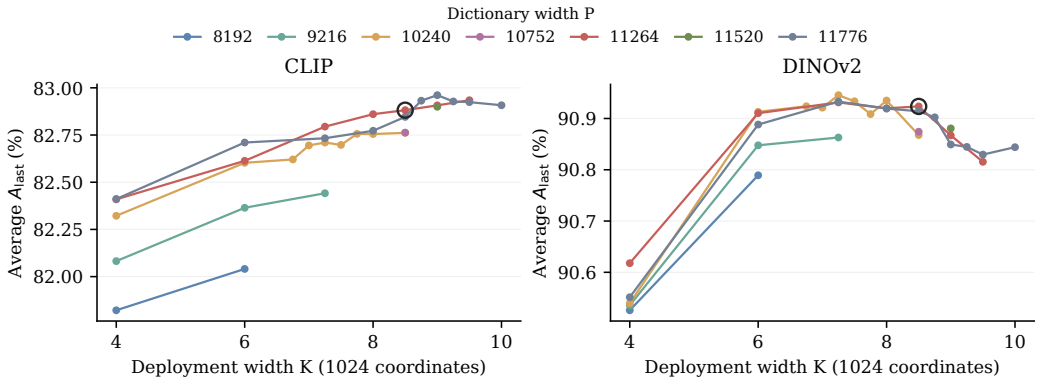


Figure 8: Average A_{last} across fixed dictionary and deployment widths. Each marker represents a measured configuration, and the outlined marker denotes the configuration used in the main tables.

Table 9: InnerCIL CLIP compute (s) and persistent state (MiB). The Total row sums the eight dataset rows.

Dataset	Feature (s)	Method (s)	Time Total (s)	Memory (MiB)
CIFAR	122.41	12.94	135.35	360.18
Aircraft	16.61	9.43	26.05	360.18
CUB	15.13	9.75	24.87	382.06
Cars	19.79	18.75	38.54	381.18
Food	171.36	14.09	185.45	360.18
IN-R	57.77	11.59	69.36	382.06
UCF	22.92	9.66	32.57	360.18
Object	62.20	11.25	73.45	382.06
Total	488.18	97.47	585.66	2968.08

For candidate width $D_P = L_{\mathcal{B}} d_h + P + 1$ and deployed width $D_K = L_{\mathcal{B}} d_h + K + 1$, the reported persistent state contains

$$\begin{aligned} &\text{raw Packed Gram:} && D_P(D_P+1)/2, \\ &\text{label cross-statistic:} && D_P C_t, \\ &\text{native class moments and counts:} && 2PC_t + C_t, \\ &\text{deployed classifier:} && D_K C_t. \end{aligned} \tag{46}$$

Registered indices require $O(P)$ metadata. The deployed support and local RMS values require $O(K)$, feature scales require $O(D_K)$, and class order requires $O(C_t)$. These deployment quantities are included in the Memory column of Table 5. The packed Gram and checkpoint metadata are class independent, whereas target statistics, class moments, and the classifier grow linearly with the number of seen classes. No term grows with the number of previously observed images.

For a task containing N_t realized training views, dense raw-statistic accumulation costs $O(N_t D_P^2)$, Fisher scoring costs $O(PC_t)$, and the selected ridge solve costs $O(D_K^3 + D_K^2 C_t)$. Attention pooling over the MLP activations already produced by the frozen ViT costs $O(MP)$ per training image and $O(MK)$ per test image after support selection. Support selection reduces the analytic solve from width D_P to D_K, while raw-statistic accumulation retains the D_P-dimensional Packed Gram.

A.6 Additional Ablations and Robustness

Post-CLS random ReLU control. The random dictionary uses the multi-depth CLS vector $h(x)$ defined in Section 3.3. Each depth contributes $P/L_{\mathcal{B}}$ random functions. For each block b, draw $\Omega_b \in \mathbb{R}^{d_h \times (P/L_{\mathcal{B}})}$ with independent $\mathcal{N}(0, 1/d_h)$ entries and keep it fixed throughout the incremental sequence. With $\Omega := \mathrm{blockdiag}_{b \in \mathcal{B}}(\Omega_b)$, the P candidate responses are $r_\Omega(x) := \mathrm{ReLU}(\Omega^\top h(x))$.

They replace the native local coordinates, while the global branch, Visual Fisher selection, RMS scaling, and ridge fitting remain unchanged.

Progressive construction with DINOv2. Table 10 reports the DINOv2 progressive construction under a shared data protocol and analytic readout.

Table 10: DINOv2 progressive component analysis. Values are A_{last} accuracy (%). Δ_{Full} is the change in Average A_{last} relative to InnerCIL.

Stage	Representation and learning system	CIFAR	Aircraft	CUB	Cars	Food	IN-100	UCF	VTAB	Average A_{last}	Δ_{Full}
1	Final-block CLS	88.00	76.45	89.07	85.33	88.25	86.64	85.76	92.27	86.47	−4.45
2	+ Multi-depth CLS	89.42	81.76	89.94	88.97	90.87	87.72	88.93	94.07	88.96	−1.96
3	+ Registered native dictionary (Random-K)	91.48	84.79	90.35	91.52	93.26	88.86	90.32	94.91	90.69	−0.24
4	**+ Visual Fisher (InnerCIL)**	**91.59**	**85.48**	**90.61**	**91.88**	**93.40**	**89.06**	**90.45**	**94.91**	**90.92**	**+0.00**

Single-module replacements with DINOv2. Table 11 applies the four single-module replacements of Table 4 to eight DINOv2 benchmarks. Each control retains the corresponding full model's data, global branch, capacity budget, and ridge coefficient.

Table 11: DINOv2 single-module replacements, using the metrics in Table 4.

Configuration	CIFAR	Aircraft	CUB	Cars	Food	IN-100	UCF	VTAB	Average A_{last}	Δ_{Full}
Post-CLS random ReLU	90.75	80.17	**90.68**	88.33	92.51	89.04	88.75	94.36	89.32	−1.60
CLS-token native neurons	90.90	82.27	90.46	89.09	92.86	88.88	88.59	94.42	89.68	−1.24
Uniform-pooled native neurons	91.49	80.41	88.85	89.58	93.04	88.38	88.99	**95.52**	89.53	−1.39
Visible-only native neurons	90.66	83.77	90.23	91.34	92.47	88.22	89.71	94.79	90.15	−0.77
InnerCIL	**91.59**	**85.48**	90.61	**91.88**	**93.40**	**89.06**	**90.45**	94.91	**90.92**	+0.00

Robustness across random seeds. We compare InnerCIL and AnaCP* using the same three random seeds under CLIP and DINOv2. Each seed determines the class-arrival order and method-specific randomness, while the data split, task size, training views, and method configuration remain fixed.

Table 12(a): Three-seed robustness under CLIP ViT-B/16. Results are mean ± sample standard deviation (%).

Dataset	InnerCIL		AnaCP*	
	A_{avg}	A_{last}	A_{avg}	A_{last}
CIFAR	**90.71 ± 0.74**	**85.26 ± 0.01**	88.07 ± 0.87	81.50 ± 0.08
Aircraft	**76.39 ± 1.07**	**70.63 ± 0.00**	70.83 ± 1.83	63.64 ± 0.12
CUB	**89.30 ± 0.32**	**83.93 ± 0.01**	89.29 ± 0.56	83.58 ± 0.12
Cars	**94.38 ± 0.41**	**90.52 ± 0.01**	92.61 ± 0.36	87.74 ± 0.14
Food	**95.49 ± 0.03**	**92.71 ± 0.00**	95.18 ± 0.08	92.22 ± 0.02
IN-R	**89.63 ± 0.32**	**84.81 ± 0.01**	89.26 ± 0.34	84.24 ± 0.21
UCF	**91.83 ± 0.63**	**89.15 ± 0.00**	90.99 ± 0.90	87.20 ± 0.06
Object	**73.96 ± 1.75**	**66.08 ± 0.03**	71.97 ± 1.49	63.04 ± 0.17
Average	**87.71**	**82.88**	86.03	80.40

With fixed samples and views, the near-invariant A_{last} in Table 12 agrees with InnerCIL's additive-statistic closure (Appendix A.3). Class order changes the intermediate class sets and recognition difficulty, explaining the larger A_{avg} variation observed for both methods.

Table 12(b): Three-seed robustness under DINOv2 ViT-B/14. Results are mean ± sample standard deviation (%).

Dataset	**InnerCIL**		**AnaCP***	
	A_{avg}	A_{last}	A_{avg}	A_{last}
CIFAR	94.67 ± 0.61	91.59 ± 0.00	**95.37 ± 0.46**	**92.16 ± 0.10**
Aircraft	**89.58 ± 0.80**	**85.49 ± 0.02**	87.73 ± 0.93	83.02 ± 0.70
CUB	94.38 ± 0.56	**90.60 ± 0.01**	**94.43 ± 0.67**	90.10 ± 0.06
Cars	**95.16 ± 0.65**	**91.87 ± 0.01**	93.35 ± 0.65	90.84 ± 0.33
Food	**96.01 ± 0.11**	**93.41 ± 0.00**	94.95 ± 0.15	92.30 ± 0.17
IN-100	**93.64 ± 0.95**	**89.06 ± 0.00**	93.63 ± 0.83	88.75 ± 0.06
UCF	**92.46 ± 0.39**	**90.45 ± 0.00**	92.19 ± 0.35	89.37 ± 0.54
VTAB	**97.58 ± 0.10**	**94.91 ± 0.00**	96.85 ± 0.09	94.30 ± 0.11
Average	**94.19**	**90.92**	93.56	90.11