CADT: CATEGORY-AWARE ADAPTATION WITH DISENTANGLED TRANSFORMATIONS FOR INCREMENTAL VISION-LANGUAGE OBJECT DETECTION

Anonymous authors
Paper under double-blind review

ABSTRACT

Incremental vision-language object detection (IVLOD) enables detectors to continuously adapt to new downstream domains while preserving pre-trained open-vocabulary generalization. However, existing methods still suffer from parameter interference during continual training. Specifically, this interference manifests in two dimensions. Regarding inter-category interference, existing category-level adaptation methods typically learn update matrices in a single parameter space, coupling the update direction and amplitude. From an optimization perspective, this coupling mechanism severely limits the degrees of freedom for category-specific adjustments. Consequently, dominant gradients from shared generic features easily overwhelm the signals of category-unique features, forcing category experts to update along generic features and resulting in severe homogenization. Regarding intra-category interference, recurring categories across tasks are updated with different task contexts, causing representation drift to accumulate task-by-task, which exacerbates forgetting. To address these issues, we propose a category-aware adaptation framework with disentangled transformations (CADT). For inter-category interference, we propose disentangled category-aware transformation (DCT). Utilizing the structure of SVD, DCT explicitly decouples the update direction and amplitude for separate optimization. This mechanism releases the degrees of freedom for parameter updates, breaks the homogenization bottleneck of optimization, and enables the independent adjustment of adaptation direction and intensity to learn category-specific features. For intra-category interference, we design category adaptation consistency (CAC), which fuses the historical and current category parameter and imposes consistency constraints to avoid excessive parameter deviation. Experiments on ODinW-13 and ODinW-O show that CADT achieves state-of-the-art performance, striking a favorable balance between category-specific adaptation and knowledge preservation.

1 INTRODUCTION

Benefiting from large-scale pre-training, vision-language object detection (VLOD) models have demonstrated strong zero-shot detection capabilities in open scenarios Li et al. (2022); Liu et al. (2024); Wang et al. (2024). However, their performance often degrades significantly in specialized downstream domains, motivating the need for continual adaptation to sequentially arriving data. This setting is referred to as incremental vision-language object detection (IVLOD) Deng et al. (2024); Cappellino et al. (2026); Wang et al. (2026). To enable efficient incremental adaptation, existing methods commonly employ parameter-efficient fine-tuning (PEFT) techniques Cao et al. (2026), such as Low-Rank Adaptation (LoRA) Hu et al. (2022), to sequentially optimize lightweight adaptation modules. While recent methods extend adaptation to the category level to enable fine-grained specialization, they still suffer from parameter interference during continual training.

Specifically, this interference manifests first as inter-category interference. To capture fine-grained visual characteristics, recent methods introduce category-specific experts for customized adjustments, as illustrated in Fig. 1(a). However, they typically learn these experts as holistic matrices within a single parameter space, inherently coupling the adaptation direction and amplitude. In object detection, different categories inevitably share foundational visual attributes (e.g., general

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.

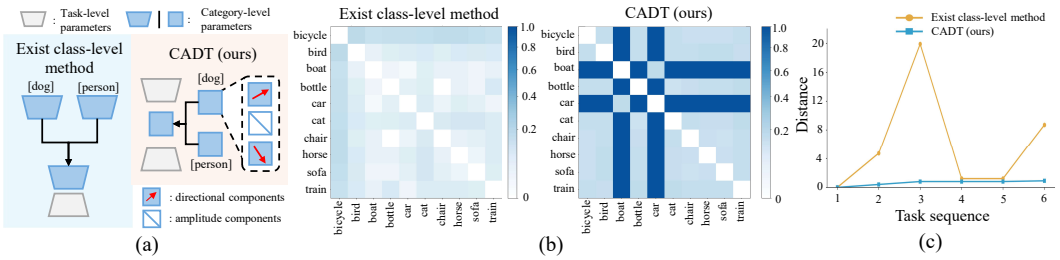


Figure 1: (a) Existing category-level methods learn holistic expert updates that couple adaptation direction and amplitude. DCT disentangles them into direction and amplitude to enhance category-specific plasticity. (b) Pairwise discrepancy heatmaps show smaller differences among category updates in existing methods and greater category diversity with DCT.(c) Across sequential tasks, categories suffer from cross-task representation drift, while CAC maintains more consistent adaptations. Detailed experimental settings and metric definitions are provided in Appendix A.6.

contours or background context), which generate overwhelmingly large gradients during backpropagation. Because the coupled parameterization lacks the degrees of freedom to independently adjust feature selection (direction) and activation strength (amplitude), these dominant gradients from shared features easily swallow the weak, fine-grained gradient of category-unique attributes. Consequently, to minimize the overall loss, the optimization process forces all category experts to update along the shared generic directions, leading to expert homogenization and a loss of category-specific plasticity, as illustrated in Fig. 1(b). To address this optimization bottleneck, we propose disentangled category-aware transformation (DCT). Inspired by SVD Golub & Reinsch (1971), DCT explicitly factorizes the category-specific transformation into independent directional and amplitude components. This structural decoupling fundamentally alters the optimization dynamics by unlocking the degrees of freedom for parameter updates. Crucially, the isolated directional parameters provide the essential structural foundation for feature subspace diversity. To explicitly realize this diversity, we introduce an intra-direction orthogonal constraint ($\mathcal{L}_d$). By actively penalizing alignment with shared generic directions, $\mathcal{L}_d$ drives the decoupled parameters to explore diverse, category-unique feature subspaces without compromising basic detection performance. Meanwhile, the independent amplitude components seamlessly preserve the activation strengths of foundational features. Through the synergy of asymmetric gradient projection and orthogonal regularization, DCT prevents unique feature signals from being overwhelmed, successfully breaking the homogenization bottleneck to learn discriminative category representations.

Furthermore, IVLOD faces intra-category interference during the continual learning process. Since recurring categories inevitably exist across sequential tasks, their corresponding parameters are repeatedly updated within different task contexts. Without an explicit maintenance mechanism, such cross-task context overwriting causes representation drift to accumulate task-by-task, as illustrated in Fig. 1(c), which significantly exacerbates forgetting. To mitigate this, we design category adaptation consistency (CAC). When updating recurring categories, CAC consolidates historical and newly learned category adaptations through SVD to extract important adaptation directions, and imposes a consistency constraint on the learning process. This direction-based fusion effectively avoids excessive parameter deviation, ensuring that previously acquired category structures are preserved while new task knowledge is seamlessly integrated.

Building upon these designs, we propose CADT, a **C**ategory-aware **A**daptation framework with **D**isentangled **T**ransformation for parameter-efficient IVLOD. CADT performs category-specific adaptation through disentangled transformations in the LoRA space and consolidates historical adaptations of recurring categories to mitigate forgetting, which comprehensively resolves both inter-category and intra-category interferences to achieve stable and fine-grained incremental adaptation. We evaluate CADT under the IVLOD setting on ODinW-13 Li et al. (2022) and ODinW-O, with the latter specifically emphasizing categories that recur across tasks. Experimental results demonstrate that CADT consistently outperforms state-of-the-art methods in both settings. Our main contributions are summarized as follows:

- We propose disentangled category-aware transformation (DCT), which decouples the update direction and amplitude to release optimization degrees of freedom, breaking the homogenization bottleneck and enabling category-unique feature learning. Through dynam-

ics analysis, we theoretically show that dense optimization inherently causes homogenization of expert matrices, whereas DCT breaks this optimization bottleneck.
- We design category adaptation consistency (CAC), which fuses historical and current parameter directions alongside consistency constraints to reduce cumulative representation drift and forgetting for recurring categories.
- Experiments on the ODinW-13 and ODinW-O benchmarks demonstrate that CADT achieves state-of-the-art performance, attaining H scores of 53.7 and 52.7, respectively, exhibiting mitigation of catastrophic forgetting and enhanced incremental adaptation.

2 Related Work

Vision-Language Object Detection. Benefiting from large-scale pre-training on image-text pairs Shao et al. (2019); Kamath et al. (2021); Plummer et al. (2015), existing vision-language object detection models have achieved strong zero-shot performance in general scenarios. Early approaches Gu et al. (2021); Zang et al. (2022) transfer CLIP Radford et al. (2021) representations to object detection through additional region-level alignment, yet still struggle with precise object localization. Recent methods can be broadly divided into two directions. The first incorporates object detection into Large Multimodal Models (LMMs)Chen et al. (2024); Bai et al. (2025) by formulating recognition and localization as autoregressive vision-language generation. Representative works in the Griffon series Zhan et al. (2024b; 2025; 2024a) progressively equip general-purpose LMMs Team et al. (2024); Touvron et al. (2023) with multi-object localization, high-resolution dense detection, and unified vision-centric and vision-language modeling. Nevertheless, these models incur substantial computational and parameter costs while still exhibiting limited open-vocabulary detection capabilities. The second direction builds specialized open-vocabulary detectors through large-scale vision-language pre-training, including GLIP Li et al. (2022), GroundingDINO Liu et al. (2024), YOLO-World Cheng et al. (2024), and OV-DINO Wang et al. (2024). These methods achieve localization with lower computational costs. However, continual adaptation may cause forgetting of previously tasks and degrade their original open-vocabulary capability.

Incremental Vision-Language Object Detection. IVLOD extends Vision-Language Object Detection to the continual learning setting, requiring models to adapt to sequentially introduced domains and categories while preserving both previously acquired knowledge and the zero-shot generalization capability. Traditional Incremental Object Detection (IOD) methods, such as CL-DETR Liu et al. (2023), TFA Wang et al. (2020), AT Houlsby et al. (2019), and iDETR Dong et al. (2023), are designed for closed-set continual detection without open-vocabulary recognition capabilities. OW-DETR Gupta et al. (2022) first introduces the open-world setting into IOD, but its generalization capability remains constrained by the closed-set model. To overcome this limitation, existing IVLOD methods build upon VLOD models and adopt Parameter-Efficient Fine-Tuning (PEFT) techniques, such as prompt tuning Cao et al. (2026), adapters Yu et al. (2024), and Low-Rank Adaptation (LoRA) Bałazy et al. (2024); Li et al. (2026), to enable efficient incremental adaptation while preserving open-vocabulary capabilities. ZiRa Deng et al. (2024) applies dual-norm regularization to the residual branches of the visual and textual encoders, and DGS Wang et al. (2026) constructs dynamic group-wise adapter subspaces for cross-task knowledge sharing. Most existing IVLOD methods perform adaptation at the task level, encoding all categories within a unified task update. DitHub Cappellino et al. (2026) is a notable exception that extends LoRA adaptation to the category level by sharing one LoRA component within each task while optimizing and maintaining category-specific component, enabling more fine-grained adaptation across categories. However, its coupled parameterization can drive category experts toward highly aligned update directions, limiting category-specific plasticity. Moreover, re-optimizing recurring categories under different task contexts may overwrite previously acquired category knowledge, leading to representation drift and catastrophic forgetting. In this work, we build upon large-scale pretrained open-vocabulary detectors to develop a parameter-efficient IVLOD framework that disentangles category-specific adaptation while maintaining consistent updates for recurring categories.

3 Methodology

3.1 Problem Setting

We consider a sequence of incremental tasks $\mathcal{T} = \{D_1, D_2, \ldots, D_T\}$, where each task D_t contains images annotated with categories from $\mathcal{C}_t$. Specifically, each sample in D_t is represented as (x, y), where x denotes an image and $y = \{(b_i, c_i)\}_{i=1}^{N}$ contains bounding boxes b_i and labels $c_i \in \mathcal{C}_t$.

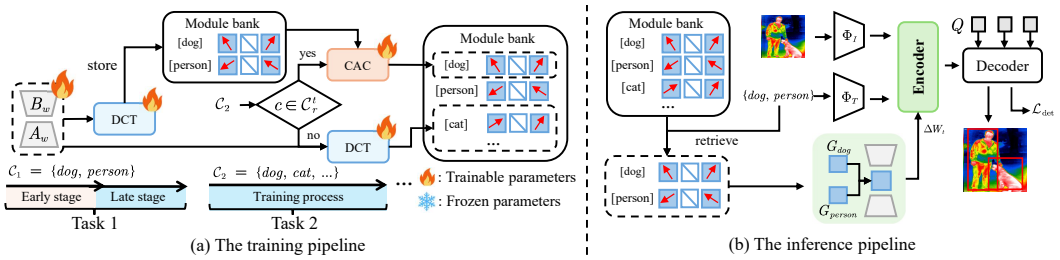


Figure 2: Training and inference pipelines of the proposed CADT framework. (a) During incremental training, CADT first learns a shared LoRA anchor and then constructs category-specific transformations using DCT, while CAC consolidates updates for recurring categories in subsequent tasks. (b) During inference, the modules corresponding to the queried categories are retrieved from the bank and combined with the LoRA anchor to construct task updates for the detector.

Unlike conventional incremental detection settings, IVLOD allows overlapping categories across tasks (i.e., $\mathcal{C}_t \cap \mathcal{C}_{t'} \neq \emptyset$). The goal of IVLOD is to adapt the model to the task D_t while retaining knowledge from previous tasks $\{D_1, \ldots, D_{t-1}\}$ and preserving zero-shot generalization on the general domain D_{ZS}.

3.2 Overview of CADT

The overall pipeline of our proposed CADT framework is depicted in Fig. 2. CADT builds upon the pre-trained GroundingDINO Liu et al. (2024), comprising an image backbone Φ_I, a text backbone Φ_T, and a transformer-based encoder-decoder detector Φ_D. Given an input image x and the category set $\mathcal{C}_t$ of task t, the Φ_I extracts visual features E_i, while the Φ_D encodes the category names into text features E_t. The detector performs vision-language fusion over the visual features, text features, and object queries Q, producing object embeddings O and bounding boxes B. Detection scores C are obtained by measuring the similarity between the object and text embeddings. The overall forward process is formulated as

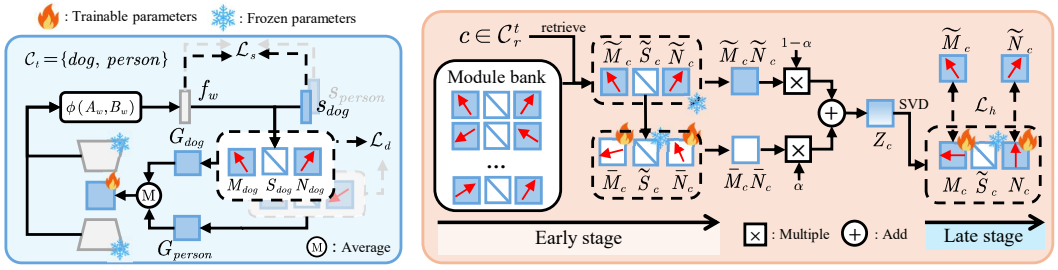
$$E_i = \Phi_I(x), \qquad E_t = \Phi_T(\mathcal{C}_t), \qquad \{O, B\} = \Phi_D(E_i, E_t, Q), \qquad C = OE_t^\top, \tag{1}$$

where $\top$ denotes the transpose operation. During training, we freeze all pre-trained parameters and insert Low-Rank Adaptation (LoRA) Hu et al. (2022) into the encoder of the detector. Given a frozen projection matrix $W_0 \in \mathbb{R}^{d\times k}$, LoRA parameterizes its update as $\Delta W = BA$, where $A \in \mathbb{R}^{r\times k}$, $B \in \mathbb{R}^{d\times r}$, and $r \ll \min(d, k)$. The adapted projection is therefore expressed as $W = W_0 + \Delta W$. As shown in Fig. 2(a), the optimization of the first task is divided into early and late stages. In the early stage, only the LoRA matrices (A_w, B_w) are optimized to capture transferable adaptation knowledge, yielding the shared LoRA anchor. In the late stage, the shared anchor is frozen to provide a stable basis for subsequent category-specific adaptation. Specifically, disentangled category-aware transformation (DCT) learns a structured transformation $G_c \in \mathbb{R}^{r\times r}$ for each category c within the frozen shared LoRA rank space, yielding the category-specific update $\Delta W_c = B_w G_c A_w$. The learned DCT transformations (e.g., dog, person) are stored as category-specific parameters in a module bank. For subsequent tasks, CADT introduces a conditional routing mechanism for incoming categories. For newly encountered categories (e.g., cat), the model applies DCT to learn novel transformations. Conversely, for recurring categories, the model routes them to CAC, which retrieves the corresponding frozen historical parameters from the module bank and consolidates them with the newly learned updates. During inference, as shown in Fig. 2(b), task t retrieves the required category-specific adaptation modules from the bank according to its category set $\mathcal{C}_t$ and averages them into a unified task-level adaptation:

$$\Delta W_t = \frac{1}{|\mathcal{C}_t|} \sum_{c\in\mathcal{C}_t} \Delta W_c = B_w \left(\frac{1}{|\mathcal{C}_t|} \sum_{c\in\mathcal{C}_t} G_c \right) A_w. \tag{2}$$

3.3 Disentangled Category-Aware Transformation

Existing category-level adaptation methods typically parameterize each category expert as a holistic update matrix within a single parameter space, where adaptation directions and their corresponding



(a) (b)

Figure 3: (a) Framework of DCT. DCT modulates the anchor descriptor to generate a amplitude matrix and combines it with directional components to construct category-specific updates. (b) Pipeline of CAC. CAC consolidates current and historical directions of recurring categories through SVD and consistency regularization.

amplitudes are inherently coupled. Under such coupled parameterization, dominant gradients induced by shared generic features can overwhelm the weaker signals associated with category-unique attributes, driving different category experts toward similar update directions and thereby limiting category-specific plasticity. To address these limitations, we introduce Disentangled Category-aware Transformation (DCT), which adopts the direction–amplitude–direction structure of SVD Golub & Reinsch (1971) to decouple category-specific adaptation directions from their corresponding amplitudes. The dynamics analysis explaining why dense optimization lead to homogenization of expert matrices and how DCT alleviates this issue is provided in Appendix A.3. For each category c, rather than directly parameterizing a dense transformation $G_c \in \mathbb{R}^{r\times r}$, DCT represents it using two directional components $(M_c, N_c) \in \mathbb{R}^{r\times r}$ and a direction-wise amplitude vector $S_c \in \mathbb{R}^r$. We adopt a category-specific orthogonal initialization for M_c and N_c that preserves the initial adaptation. Since these directions are defined within the shared LoRA rank space, learning their amplitudes independently may overlook the transferable structure inherited from the anchor. We therefore generate the amplitudes by combining anchor statistics with a category-specific modulation vector $s_c \in \mathbb{R}^r$. Specifically, we summarize the rank-wise structure of the shared anchor using

$$f_w = \phi(A_w, B_w), \quad \phi(X, Y) = \mu_{\text{in}}(X) + \mu_{\text{out}}(Y), \tag{3}$$

where $\mu_{\text{in}}(\cdot)$ and $\mu_{\text{out}}(\cdot)$ denote mean pooling along the input and output dimensions, respectively. The learnable vector s_c then modulates this descriptor according to the adaptation requirements of category c, producing anchor-aware category-specific amplitudes:

$$S_c = \text{Diag}(\exp(\text{RMSNorm}(f_w \odot s_c))). \tag{4}$$

Here, $\text{RMSNorm}(\cdot)$ denotes root-mean-square normalization, and $\text{Diag}(\cdot)$ converts these coefficients into a diagonal matrix. The resulting category-specific update is formulated as

$$\Delta W_c = B_w M_c S_c N_c A_w, \qquad G_c = M_c S_c N_c. \tag{5}$$

M_c and N_c control the adaptation directions, while S_c controls their corresponding amplitudes. The learned parameters (M_c, N_c, S_c) are then stored in the module bank. Such disentangled parameterization enables more flexible category-specific adaptation, mitigating update homogenization and improving category-specific plasticity.

3.4 Category Adaptation Consistency

Categories may recur across incremental tasks and are therefore repeatedly optimized under different task contexts. Such context-dependent updates can overwrite previously acquired category knowledge, causing representation drift to accumulate across tasks and exacerbating catastrophic forgetting. To address this intra-category interference, we propose category adaptation consistency (CAC), which consolidates historical and current directions while explicitly constraining their cross-task consistency. For unseen categories, new DCT modules are initialized and optimized normally. For each recurring category $c \in \mathcal{C}_r^t$, where $\mathcal{C}_r^t = \mathcal{C}_t \cap \mathcal{C}_{1:t-1}$, let $(\widetilde{M}_c, \widetilde{N}_c)$ denote the historical directions retrieved from the module bank, and $(\bar{M}_c, \bar{N}_c)$ denote the directions learned from the current task in the early stage. A straightforward strategy Cappellino et al. (2026) is to directly consolidate the historical and current directions:

$$M_c = (1-\alpha)\widetilde{M}_c + \alpha\bar{M}_c, \qquad N_c = (1-\alpha)\widetilde{N}_c + \alpha\bar{N}_c, \tag{6}$$

where $\alpha \in [0,1]$ controls the contribution of the current adaptation. However, directly combining the historical and current directions may distort previously learned representations by mixing newly acquired information with historical directional structures. We therefore jointly consolidate the historical and current directional compositions through SVD:

$$Z_c = (1-\alpha)\widetilde{M}_c\widetilde{N}_c + \alpha\bar{M}_c\bar{N}_c = U_c\Sigma_c V_c^\top, \quad M_c \leftarrow U_{c,k}\Sigma_{c,k}^{\frac{1}{2}}, \quad N_c \leftarrow \Sigma_{c,k}^{\frac{1}{2}}V_{c,k}^\top, \tag{7}$$

where $k = r$ denotes the retained rank. To further improve category update consistency across tasks, we encourage each direction to remain consistent with its frozen historical counterpart:

$$\mathcal{L}_{\mathrm{h}} = \frac{1}{|\mathcal{C}_r^t|}\sum_{c\in\mathcal{C}_r^t}\frac{1}{2}\left[\mathrm{SmoothL1}\Big(M_c, \widetilde{M}_c\Big) + \mathrm{SmoothL1}\Big(N_c, \widetilde{N}_c\Big)\right]. \tag{8}$$

During training, this constraint is applied only to recurring categories after directional consolidation. By preserving historical directions while incorporating newly acquired task information, CAC maintains category adaptation consistency across tasks, thereby reducing representation drift and mitigating forgetting.

3.5 Optimization Objectives of CADT

Our optimization objective is based on GroundingDINO Liu et al. (2024). The detection loss $\mathcal{L}_{\mathrm{det}}$ includes L_1 and GIoU losses Rezatofighi et al. (2019) for box regression and a contrastive classification loss Lin et al. (2017) between object queries and text tokens. In addition to the update-consistency constraint $\mathcal{L}_{\mathrm{h}}$ introduced in CAC, we further employ two auxiliary objectives: the intra-direction orthogonality loss $\mathcal{L}_{\mathrm{d}}$ and the amplitude regularization loss $\mathcal{L}_{\mathrm{s}}$. The intra-direction orthogonality loss $\mathcal{L}_{\mathrm{d}}$ regularizes M_c and N_c to encourage diverse and non-redundant directional components:

$$\mathcal{L}_{\mathrm{d}} = \frac{1}{|\mathcal{C}_t|}\sum_{c\in\mathcal{C}_t}\left[\left\|M_c^\top M_c - I\right\|_F^2 + \left\|N_c N_c^\top - I\right\|_F^2\right], \tag{9}$$

where I denotes the identity matrix and $\|\cdot\|_F$ denotes the Frobenius norm. The amplitude regularization loss $\mathcal{L}_{\mathrm{s}}$ regularizes the trainable category-specific modulation vector s_c to preserve the rank-wise amplitude pattern inherited from the shared anchor. Given the anchor descriptor $f_w = \phi(A_w, B_w)$, the category-specific modulation vector s_c produces the pre-normalized amplitude representation $z_c = f_w \odot s_c$. We encourage this category-specific amplitude pattern to remain aligned with the anchor descriptor:

$$\mathcal{L}_{\mathrm{s}} = \frac{1}{|\mathcal{C}_t|}\sum_{c\in\mathcal{C}_t}\left[1 - \frac{z_c^\top f_w}{\|z_c\|_2\|f_w\|_2}\right]. \tag{10}$$

Together with the CAC update-consistency constraint $\mathcal{L}_{\mathrm{h}}$, the overall objective is formulated as

$$\mathcal{L} = \mathcal{L}_{\mathrm{det}} + \lambda_{\mathrm{d}}\mathcal{L}_{\mathrm{d}} + \lambda_{\mathrm{s}}\mathcal{L}_{\mathrm{s}} + \lambda_{\mathrm{h}}\mathcal{L}_{\mathrm{h}}, \tag{11}$$

where λ_{d}, λ_{s}, and λ_{h} denote the weighting coefficients for the directional orthogonality, amplitude regularization, and historical update-consistency objectives, respectively.

4 Experiments

Datasets. We evaluate our method on the ODinW-13 Li et al. (2022) benchmark, which contains 13 diverse subtasks covering various domains and object categories. To evaluate category recurrence and module reuse, we further introduce ODinW-O, a subset of ODinW35 with higher category overlap, where most tasks contain categories observed in previous tasks. Following the IVLOD setting, all tasks are trained sequentially and evaluated after completing the entire sequence. The retained zero-shot generalization capability is measured on COCO Lin et al. (2014). By default, ablation studies are conducted on ODinW-13, with further details provided in the appendix.

Baselines. We compare CADT with representative methods from both incremental object detection (IOD) and incremental vision-language object detection (IVLOD). For IOD baselines, CL-DETR Liu et al. (2023) and OW-DETR Gupta et al. (2022) are DETR-based continual detection methods, while TFA Wang et al. (2020), AT Houlsby et al. (2019), and iDETR Dong et al. (2023) investigate few-shot and parameter-efficient adaptation. For IVLOD baselines, we compare with ZiRa Deng et al. (2024), DitHub Cappellino et al. (2026), and DGS Wang et al. (2026), which introduce lightweight adaptation strategies.

Table 1: Comparison of mAP on ODinW-13 under different shot settings.

Shots	Method	ZCOCO	Avg	H	Ae	Aq	Co	Eg	Mu	Pa	Pv	Pi	Po	Ra	Sh	Th	Ve
0	G-DINO	47.4	47.0	47.2	19.1	20.8	64.8	60.0	25.3	56.3	54.8	65.9	22.1	62.0	32.9	70.4	57.1
5	TFA	26.4	45.8	33.5	21.9	22.3	67.4	60.7	30.6	53.6	46.8	63.6	26.9	56.3	28.0	64.3	52.5
	iDETR	43.5	51.7	47.2	25.7	25.5	70.4	63.0	50.0	50.5	54.9	64.8	**33.2**	57.6	**42.4**	**76.5**	56.9
	AT	43.7	47.2	45.4	14.6	25.0	66.6	64.2	38.9	42.0	55.5	**65.2**	27.5	52.7	32.2	71.2	57.5
	ZiRa	45.4	51.8	48.4	24.2	26.4	71.3	64.0	49.5	**60.5**	60.1	64.4	32.2	56.3	40.6	69.2	54.3
	DitHub	**45.9**	52.9	**49.2**	**26.8**	28.9	68.5	60.3	51.9	55.1	60.4	59.4	**33.2**	**68.1**	38.1	75.0	61.3
	CADT (ours)	45.8	**53.1**	**49.2**	24.5	**29.1**	**75.1**	**65.3**	**52.4**	57.3	**61.6**	62.9	29.7	65.2	32.4	72.7	**61.6**
10	TFA	34.4	46.6	39.6	21.2	22.2	66.8	60.6	32.4	50.2	55.5	65.0	27.6	57.4	28.1	66.8	52.1
	iDETR	43.5	53.3	47.9	25.4	27.7	65.6	**67.6**	48.0	60.2	56.3	63.9	35.2	59.4	**53.6**	74.7	55.1
	AT	43.1	47.3	45.1	18.7	25.4	69.8	66.3	35.8	48.3	53.4	64.1	28.9	50.3	32.5	66.6	55.4
	ZiRa	46.1	53.2	49.4	25.6	26.0	71.7	65.3	48.2	**62.8**	61.2	**65.5**	35.6	55.7	46.9	71.0	56.1
	DitHub	46.5	54.4	50.1	**26.2**	**30.3**	68.1	65.7	**52.8**	56.8	**63.4**	61.9	**36.7**	66.8	41.5	**76.0**	61.5
	CADT (ours)	**46.6**	**54.5**	**50.2**	23.9	29.9	**76.6**	65.6	51.1	57.9	62.9	60.8	35.9	**68.6**	41.0	71.1	**63.2**
Full	TFA	31.0	47.9	37.6	23.8	30.7	67.2	61.8	30.5	50.2	47.7	60.9	29.3	61.7	31.4	66.2	61.6
	iDETR	37.3	58.7	45.6	32.6	46.7	71.0	68.6	**55.3**	58.9	64.5	71.0	50.3	63.3	39.2	77.1	64.8
	AT	42.3	51.1	46.3	23.6	39.9	72.3	65.5	31.5	50.5	60.5	66.1	39.1	53.5	34.0	68.1	60.2
	OW-DETR	31.2	55.6	40.0	28.5	43.8	70.5	67.8	43.8	56.8	63.1	69.5	45.2	59.0	37.0	74.4	63.2
	CL-DETR	32.2	57.3	41.2	29.4	45.2	71.9	69.9	45.2	58.5	65.1	71.7	46.6	60.8	38.1	76.7	65.2
	ZiRa	46.3	58.0	51.5	31.8	47.4	71.8	64.7	46.5	62.7	66.4	71.0	48.5	63.0	41.4	76.1	62.4
	DitHub	**47.0**	62.2	53.5	34.6	50.7	70.5	68.6	49.3	**65.6**	69.6	71.1	**56.7**	70.9	**52.8**	79.3	68.2
	DGS	46.4	60.9	52.7	**40.3**	**54.8**	60.5	**78.3**	42.6	59.1	58.4	**75.9**	56.1	69.5	50.5	79.4	66.4
	CADT (ours)	**47.0**	**62.5**	**53.7**	37.6	45.8	**74.0**	72.3	54.5	62.7	**70.2**	71.0	51.9	**77.1**	45.6	**81.2**	**68.5**

Metrics. We report the average mAP Dave et al. (2021) after sequentially training on all tasks, denoted as Avg, where mAP is calculated as the mean $\mathrm{AP}_{50\text{–}95}$ over all categories. To evaluate the trade-off between continual adaptation and preservation of open-vocabulary generalization, we adopt the harmonic mean Xian et al. (2017):

$$\mathrm{H} = \frac{2 \times (\mathrm{ZCOCO} \times \mathrm{Avg})}{\mathrm{ZCOCO} + \mathrm{Avg}}, \tag{12}$$

where ZCOCO denotes the mAP on zero-shot COCO. To measure forgetting, we further introduce zero-shot forgetting Δ_{ZS} and average task forgetting $\bar{F}$ Dave et al. (2021). Specifically, Δ_{ZS} denotes the degradation in COCO zero-shot performance after sequentially learning all tasks. The average task forgetting is defined as

$$\bar{F} = \frac{1}{T-1} \sum_{i=1}^{T-1} \left(\max_{t \in \{i,\ldots,T-1\}} R_{t,i} - R_{T,i} \right), \tag{13}$$

where $R_{t,i}$ denotes the mAP on task i after learning task t. Lower Δ_{ZS} and $\bar{F}$ indicate less forgetting.

Implementation Details. We adopt GroundingDINO-T (G-DINO) Liu et al. (2024) as the base detector, initialized with weights pre-trained on Objects365 Shao et al. (2019), GoldG Li et al. (2022), and Cap4M Li et al. (2022). For our method, LoRA modules are inserted into the encoder layers. During training, all pre-trained parameters are frozen, and only LoRA parameters with rank $r = 16$ are optimized. We use AdamW Loshchilov & Hutter with an initial learning rate of 4×10^{-3} and weight decay of 10^{-2}. The transition point between independent category adaptation and CAC consolidation is set to 70% of the training process. The CAC merging coefficient is set to $\alpha = 0.9$, and the loss weights are set to $\lambda_{\mathrm{d}} = \lambda_{\mathrm{s}} = \lambda_{\mathrm{h}} = 0.1$.

4.1 Main Results

Performance on ODinW-13. As shown in Table 1, our proposed CADT achieves strong overall performance across all evaluated shot settings, attaining the highest Avg and H in every setting. In the full-data setting, CADT achieves substantial improvements over conventional IOD methods such as iDETR Dong et al. (2023) and CL-DETR Liu et al. (2023). Specifically, compared with iDETR, CADT improves ZCOCO, Avg, and H by 9.7, 3.8, and 8.1 points, respectively. This stark contrast highlights that conventional incremental learning techniques are inadequate for balancing downstream task adaptation with the preservation of pre-trained open-vocabulary knowledge. Furthermore, compared with existing IVLOD methods, CADT achieves the best overall performance, reaching 47.0 ZCOCO, 62.5 Avg, and 53.7 H. CADT also remains effective under data-scarce settings. Across the 5-shot and 10-shot settings, CADT consistently achieves the best Avg and the best H. Compared with the second-best method, DitHub Cappellino et al. (2026), CADT improves Avg by 0.2 and 0.1 points under 5-shot and 10-shot settings, respectively, while matching the best 5-shot H and improving the 10-shot H by 0.1 points. These results conclusively validate the efficacy of CADT in achieving precise adaptation while mitigating catastrophic forgetting.

Table 2: Comparison of mAP on ODinW-O.

Method	ZCOCO	Avg	H	Ae	Hw	Pv	Sd	Th	Ve
G-DINO	**47.4**	53.2	50.1	45.1	67.5	58.1	25.8	70.4	51.9
iDETR	37.4	59.0	45.8	49.7	67.8	67.7	36.0	**78.4**	54.6
AT	44.9	56.9	50.2	46.9	65.8	67.3	30.6	76.0	**54.8**
ZiRa	44.4	57.6	50.1	39.9	**68.0**	64.9	**46.3**	77.3	49.5
DitHub	47.2	59.2	52.5	49.2	66.9	**72.9**	38.7	74.7	52.9
CADT (ours)	47.0	**59.9**	**52.7**	**50.5**	**68.0**	72.8	42.3	72.8	52.9

Table 3: Comparison of catastrophic forgetting.

Method	ODinW-13		ODinW-O	
	$\Delta_{\mathrm{ZS}}\downarrow$	$\bar{F}\downarrow$	$\Delta_{\mathrm{ZS}}\downarrow$	$\bar{F}\downarrow$
ZiRa	1.1	4.8	3.0	4.3
DitHub	0.4	2.8	**0.2**	6.1
CADT (ours)	0.4	0.1	**0.2**	**3.4**

Performance on ODinW-O. To further evaluate CADT under increased category overlap, we conduct experiments on ODinW-O. As shown in Table 2, our method achieves the highest Avg and H of 59.9 and 52.7, respectively, while retaining 47.0 ZCOCO. Compared with DitHub Cappellino et al. (2026), CADT improves Avg by 0.7 points and H by 0.2 points. Although other methods perform better on individual tasks, CADT achieves the strongest overall balance between incremental adaptation and zero-shot generalization.

Forgetting on ODinW-13 and ODinW-O. Furthermore, as shown in Table 3, CADT exhibits low forgetting on both benchmarks. On ODinW-13, CADT achieves the lowest Δ_{ZS} of 0.4 and average forgetting $\bar{F}$ to 0.1, a 2.7-point reduction compared with DitHub Cappellino et al. (2026). More notably, on the challenging ODinW-O dataset characterized by severe category overlap, our method still demonstrates superior stability. CADT again achieves the lowest Δ_{ZS} of 0.2 and the lowest $\bar{F}$ of 3.4, reducing average forgetting by 2.7 points compared with DitHub at 6.1. These consistent results across varying task setups confirm that our strategy effectively mitigates parameter interference.

4.2 Model Analysis and Ablation Studies

Effectiveness of different components. We conduct progressive ablation on the key components of CADT, including DCT, CAC, and the three auxiliary losses. As shown in Table 4, compared with the baseline, which directly learns a complete pair of LoRA matrices for each category, introducing DCT improves Avg by 2.2 points and H by 0.5 points, while reducing $\bar{F}$ from 4.5 to 0.6. This demonstrates that disentangling adaptation direction and amplitude facilitates category-specific specialization and alleviates interference among category updates. Adding $\mathcal{L}_{\mathrm{d}}$ and $\mathcal{L}_{\mathrm{s}}$ further improves Avg and H by 0.9 and 0.4 points, respectively, with $\bar{F}$ remaining at 0.6, confirming the benefit of explicitly regularizing the disentangled directional and amplitude components. Introducing CAC without $\mathcal{L}_{\mathrm{h}}$ subsequently decreases Avg and H by 0.7 and 0.6 points, respectively, while increasing $\bar{F}$ to 0.8, indicating that directional consolidation alone is insufficient to maintain consistent adaptations across tasks. Finally, incorporating $\mathcal{L}_{\mathrm{h}}$ brings improvements of 1.8 and 0.8 points in Avg and H, respectively, while reducing $\bar{F}$ to 0.1 , demonstrating the importance of explicitly constraining recurring-category drift. Combining all components achieves the highest Avg and H, outperforming the baseline by 4.2 and 1.1 points, respectively, while reducing $\bar{F}$ by 4.4 points.

Table 4: Component ablation of CADT.

DCT	CAC	$\mathcal{L}_{\mathrm{d}}$	$\mathcal{L}_{\mathrm{s}}$	$\mathcal{L}_{\mathrm{h}}$	Avg	H	$\bar{F}\downarrow$
					58.3	52.6	4.5
✓					60.5	53.1	0.6
✓		✓			61.0	53.2	0.6
✓		✓	✓		61.4	53.5	0.6
✓	✓	✓	✓		60.7	52.9	0.8
✓	✓	✓	✓	✓	**62.5**	**53.7**	**0.1**

Effect of parameterization designs in DCT. We compare different parameterization designs in DCT in Table 5. The M_c-only variant corresponds to direct G_c, which learns the category-specific transformation as a single dense matrix and achieves an Avg of 58.3 and an H of 52.0. This dense parameterization couples adaptation directions with their amplitude, limiting their independent adjustment and potentially biasing the update toward shared dominant directions. Introducing both M_c and N_c improves Avg to 60.0 and H to 52.4 by enabling more flexible directional recomposition. However, without S_c, it lacks explicit direction-wise amplitude modulation. Combining M_c, N_c, and S_c achieves the best ZCOCO, Avg, and H of 47.3, 61.0, and 53.3, respectively. Compared with directly learning a single dense G_c, the disentangled parameterization improves Avg by 2.7 points and H by 1.3 points, demonstrating the effectiveness of decoupling adaptation direction and amplitude for category-specific adaptation.

Table 5: Ablation study of different parameterization designs in DCT.

M_c	N_c	S_c	ZCOCO	Avg	H
✓			46.9	58.3	52.0
✓	✓		46.5	60.0	52.4
✓	✓	✓	47.3	**61.0**	**53.3**

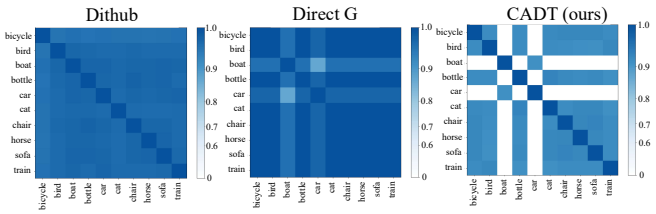


Figure 4: Pairwise similarity of category-specific adaptation updates on Pv.

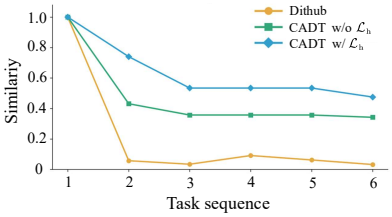


Figure 5: Effect of CAC on category adaptation consistency across tasks.

Comparison of consolidate strategies in CAC. We compare five strategies for consolidating recurring categories in Table 6. Direct separately consolidates the historical and current M_c and N_c through direct addition without SVD. The M_c and N_c strategies apply SVD-based consolidation to one directional matrix while retaining the historical version of the other. The G_c strategy consolidates M_c and N_c and applies the consistency constraint to the complete transformation G_c. The M_c, N_c strategy is adopted in our method. Direct yields the lowest Avg and H of 56.1 and 51.2 and the highest $\bar{F}$ of 3.4, suggesting that weighted addition without SVD is insufficient to preserve the joint directional structure. Applying SVD-based consolidation to only M_c or N_c improves Avg to 60.7 or 61.0 and reduces $\bar{F}$ to 1.2 or 0.8, respectively, indicating that preserving either directional factor already benefits cross-task consistency. Constraining the composed transformation G_c after directional consolidation further improves Avg and H to 61.4 and 53.3, while reducing $\bar{F}$ to 0.4. However, since G_c jointly mixes directional and amplitude information, consistency imposed in the composed space may not directly preserve each directional factor. Finally, separately constraining the consolidated M_c and N_c achieves the highest Avg of 62.5 and H of 53.7, together with the lowest $\bar{F}$ of 0.1, showing that factor-wise consistency more effectively preserves historical directional structures while accommodating new task knowledge.

Table 6: Comparison of different consolidate strategies in CAC.

Strategy	ZCOCO	Avg	H	$\bar{F}\downarrow$
Direct	47.1	56.1	51.2	3.4
M_c	47.1	60.7	53.0	1.2
N_c	46.7	61.0	52.9	0.8
G_c	47.1	61.4	53.3	0.4
M_c, N_c	47.0	**62.5**	**53.7**	**0.1**

Visualization of category update diversity. We visualize the pairwise CKA similarities between category-specific adaptation updates on Pv in Fig. 4 (see Appendix A.6.1 for the CKA definition). DitHub exhibits highly similar category updates, revealing substantial update homogenization. Directly learning a dense G_c further increases the similarities, indicating that coupling direction and amplitude aggravates this homogenization. In contrast, CADT produces substantially lower and more diverse similarities across categories, demonstrating that disentangling direction and amplitude effectively promotes category-specific update diversity.

Visualization of category adaptation consistency. We compare each recurring category update with its initially learned state throughout the task sequence in Fig. 5 (see Appendix A.6.2 for the consistency metric). DitHub exhibits relatively low CKA similarity across tasks, indicating substantial degradation of category adaptation consistency. CADT without $\mathcal{L}_h$ noticeably improves the similarity, showing that directional consolidation alleviates cross-task drift, but the consistency remains limited. With $\mathcal{L}_h$, CADT achieves the highest similarity throughout the task sequence, yielding the best category adaptation consistency across tasks.

5 Conclusion

In this paper, we propose CADT, a parameter-efficient category-aware adaptation framework with disentangled transformations for Incremental Vision-Language Object Detection. CADT enables fine-grained category-specific adaptation while preserving previously acquired knowledge and pre-trained open-vocabulary generalization. By combining DCT, which disentangles adaptation direction and amplitude to alleviate category-update homogenization, with CAC, which consolidates historical and current adaptation directions to maintain consistency for recurring categories, CADT effectively mitigates both inter-category and intra-category interference during continual adaptation. Extensive experiments on ODinW-13 and ODinW-O demonstrate that CADT achieves a favorable balance between incremental adaptation and knowledge preservation.

REFERENCES

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. Qwen3-vl technical report. *arXiv preprint*, 2025.

Klaudia Bałazy, Mohammadreza Banaei, Karl Aberer, and Jacek Tabor. Lora-xs: Low-rank adaptation with extremely small number of parameters. In *Proceedings of the 28th European Conference on Artificial Intelligence*, pp. 3194–3201. SAGE Publications 1 Oliver's Yard, 55 City Road, London, EC1Y 1SP, 2024.

Weihao Cao, Runqi Wang, Xiaoyue Duan, Jinchao Zhang, Ang Yang, and Liping Jing. Parameter-efficient semantic augmentation for enhancing open-vocabulary object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20367–20376, 2026.

Chiara Cappellino, Gianluca Mancusi, Matteo Mosconi, Angelo Porrello, Simone Calderara, and Rita Cucchiara. Dithub: A modular framework for incremental open-vocabulary object detection. *Adv. Neural Inform. Process. Syst.*, 38:22898–22926, 2026.

Zhe Chen, Weiyun Wang, Yue Cao, Yangzhou Liu, Zhangwei Gao, Erfei Cui, Jinguo Zhu, Shenglong Ye, Hao Tian, Zhaoyang Liu, et al. Expanding performance boundaries of open-source multimodal models with model, data, and test-time scaling. *arXiv preprint*, 2024.

Tianheng Cheng, Lin Song, Yixiao Ge, Wenyu Liu, Xinggang Wang, and Ying Shan. Yolo-world: Real-time open-vocabulary object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 16901–16911, 2024.

Achal Dave, Piotr Dollár, Deva Ramanan, Alexander Kirillov, and Ross Girshick. Evaluating large-vocabulary object detectors: The devil is in the details. *arXiv preprint*, 2021.

Jieren Deng, Haojian Zhang, Kun Ding, Jianhua Hu, Xingxuan Zhang, and Yunkuan Wang. Zero-shot generalizable incremental learning for vision-language object detection. *Adv. Neural Inform. Process. Syst.*, 37:136679–136700, 2024.

Na Dong, Yongqiang Zhang, Mingli Ding, and Gim Hee Lee. Incremental-detr: Incremental few-shot object detection via self-supervised learning. In *Proceedings of the AAAI conference on artificial intelligence*, volume 37, pp. 543–551, 2023.

Gene H Golub and Christian Reinsch. Singular value decomposition and least squares solutions. In *Linear algebra*, pp. 134–151. Springer, 1971.

Xiuye Gu, Tsung-Yi Lin, Weicheng Kuo, and Yin Cui. Open-vocabulary object detection via vision and language knowledge distillation. *arXiv preprint*, 2021.

Akshita Gupta, Sanath Narayan, KJ Joseph, Salman Khan, Fahad Shahbaz Khan, and Mubarak Shah. Ow-detr: Open-world detection transformer. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 9235–9244, 2022.

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

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Liang Wang, Weizhu Chen, et al. Lora: Low-rank adaptation of large language models. In *Int. Conf. Learn. Represent.*, 2022.

Aishwarya Kamath, Mannat Singh, Yann LeCun, Gabriel Synnaeve, Ishan Misra, and Nicolas Carion. Mdetr-modulated detection for end-to-end multi-modal understanding. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1780–1790, 2021.

Liunian Harold Li, Pengchuan Zhang, Haotian Zhang, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, et al. Grounded language-image pre-training. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 10965–10975, 2022.

Shiwei Li, Xiandi Luo, Haozhao Wang, Xing Tang, Ziqiang Cui, Dugang Liu, Yuhua Li, Xiuqiang He, and Ruixuan Li. Beyond higher rank: Token-wise input-output projections for efficient low-rank adaptation. *Adv. Neural Inform. Process. Syst.*, 38:136414–136435, 2026.

Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pp. 740–755. Springer, 2014.

Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. Focal loss for dense object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2980–2988, 2017.

Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Qing Jiang, Chunyuan Li, Jianwei Yang, Hang Su, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. In *European conference on computer vision*, pp. 38–55. Springer, 2024.

Yaoyao Liu, Bernt Schiele, Andrea Vedaldi, and Christian Rupprecht. Continual detection transformer for incremental object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 23799–23808, 2023.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *Int. Conf. Learn. Represent.*

Bryan A Plummer, Liwei Wang, Chris M Cervantes, Juan C Caicedo, Julia Hockenmaier, and Svetlana Lazebnik. Flickr30k entities: Collecting region-to-phrase correspondences for richer image-to-sentence models. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 2641–2649, 2015.

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.

Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union: A metric and a loss for bounding box regression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 658–666, 2019.

Shuai Shao, Zeming Li, Tianyuan Zhang, Chao Peng, Gang Yu, Xiangyu Zhang, Jing Li, and Jian Sun. Objects365: A large-scale, high-quality dataset for object detection. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 8430–8439, 2019.

Gemma Team, Morgane Riviere, Shreya Pathak, Pier Giuseppe Sessa, Cassidy Hardin, Surya Bhupatiraju, Léonard Hussenot, Thomas Mesnard, Bobak Shahriari, Alexandre Ramé, et al. Gemma 2: Improving open language models at a practical size. *arXiv preprint*, 2024.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, et al. Llama 2: Open foundation and fine-tuned chat models. *arXiv preprint*, 2023.

Hao Wang, Pengzhen Ren, Zequn Jie, Xiao Dong, Chengjian Feng, Yinlong Qian, Lin Ma, Dongmei Jiang, Yaowei Wang, Xiangyuan Lan, et al. Ov-dino: Unified open-vocabulary detection with language-aware selective fusion. *arXiv preprint*, 2024.

Xin Wang, Thomas E Huang, Trevor Darrell, Joseph E Gonzalez, and Fisher Yu. Frustratingly simple few-shot object detection. In *International Conference on Machine Learning*, pp. 9919–9928. PMLR, 2020.

Xu Wang, Zihan Lin, Yixin Zhang, and Zilei Wang. Boosting vision-language models towards cross-domain incremental object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 6249–6260, 2026.

Yongqin Xian, Bernt Schiele, and Zeynep Akata. Zero-shot learning-the good, the bad and the ugly. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 4582–4591, 2017.

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*, pp. 23219–23230, 2024.

Yuhang Zang, Wei Li, Kaiyang Zhou, Chen Huang, and Chen Change Loy. Open-vocabulary detr with conditional matching. In *European conference on computer vision*, pp. 106–122. Springer, 2022.

Yufei Zhan, Hongyin Zhao, Yousong Zhu, Fan Yang, Ming Tang, and Jinqiao Wang. Griffon-g: Bridging vision-language and vision-centric tasks via large multimodal models. *arXiv preprint*, 2024a.

Yufei Zhan, Yousong Zhu, Zhiyang Chen, Fan Yang, Ming Tang, and Jinqiao Wang. Griffon: Spelling out all object locations at any granularity with large language models. In *European conference on computer vision*, pp. 405–422. Springer, 2024b.

Yufei Zhan, Shurong Zheng, Yousong Zhu, Hongyin Zhao, Fan Yang, Ming Tang, and Jinqiao Wang. Griffon v2: Advancing multimodal perception with high-resolution scaling and visual-language co-referring. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 22947–22957. IEEE, 2025.

A Appendix

A.1 AI use statement

AI tools were used solely to assist with language editing and polishing during the preparation of this manuscript. All technical content, methodology, experiments, analyses, and conclusions were developed and verified by the authors.

A.2 Additional Experimental Implementation

For ODinW-13, we follow the training protocol of ZiRa Deng et al. (2024), while for ODinW-O, we follow DitHub Cappellino et al. (2026). All models are trained with a batch size of 4. All experiments use a fixed random seed of 42 and are conducted on a single NVIDIA L20 GPU. GPU memory consumption ranges from 5GB to 43GB depending on the training stage and task complexity, and the complete training process takes approximately 24 hours. Our implementation is based on Python 3.9.25, PyTorch 2.4.0 with CUDA 12.4, and Detectron2 0.6.

A.3 Analysis Of the G Decomposition

We analyze how different parameterizations of the category-specific LoRA transformation affect the evolution of its spectral structure. Specifically, we show that although both direct optimization and SVD-based optimization have equivalent representation capacity, their induced optimization dynamics on the singular values are fundamentally different, which leads to different spectral distributions and consequently different expert similarities.

Spectral decomposition of expert similarity. For each category c, the learned LoRA transformation is denoted as $G_c \in \mathbb{R}^{r\times r}$. Since any matrix admits an exact singular value decomposition (SVD), we have

$$G_c = M_c \Sigma_c N_c^\top, \tag{14}$$

where $M_c, N_c \in \mathbb{R}^{r\times r}$ are orthogonal matrices satisfying

$$M_c^\top M_c = I, \qquad N_c^\top N_c = I, \tag{15}$$

and

$$\Sigma_c = \mathrm{diag}(\sigma_{c,1}, \sigma_{c,2}, \cdots, \sigma_{c,r}) \tag{16}$$

contains the singular values satisfying

$$\sigma_{c,1} \geq \sigma_{c,2} \geq \cdots \geq \sigma_{c,r} \geq 0. \tag{17}$$

Consider two expert matrices G_i and G_j. Their Frobenius inner product is

$$\langle G_i, G_j \rangle_F = \mathrm{Tr}(G_i^\top G_j). \tag{18}$$

Substituting their SVD forms gives

$$\begin{aligned} \langle G_i, G_j \rangle_F &= \mathrm{Tr}(N_i \Sigma_i M_i^\top M_j \Sigma_j N_j^\top) \\ &= \mathrm{Tr}(\Sigma_i M_i^\top M_j \Sigma_j N_j^\top N_i). \end{aligned} \tag{19}$$

Let

$$A_{ij} = M_i^\top M_j, \qquad B_{ij} = N_i^\top N_j, \tag{20}$$

where A_{ij} and B_{ij} measure the overlap between the left and right singular directions of two experts, respectively.

Then

$$\langle G_i, G_j \rangle_F = \mathrm{Tr}(\Sigma_i A_{ij} \Sigma_j B_{ij}^\top). \tag{21}$$

Expanding the trace operation yields

$$\langle G_i, G_j \rangle_F = \sum_{p=1}^{r} \sum_{q=1}^{r} \sigma_{i,p} \sigma_{j,q} (A_{ij})_{pq} (B_{ij})_{pq}. \tag{22}$$

Since orthogonal transformations preserve the Frobenius norm,

$$\|G_c\|_F = \|\Sigma_c\|_F, \tag{23}$$

the cosine similarity between two experts can be written as

$$\rho_{ij} = \frac{\sum_{p,q} \sigma_{i,p} \sigma_{j,q} (M_i^\top M_j)_{pq} (N_i^\top N_j)_{pq}}{\|\Sigma_i\|_F \|\Sigma_j\|_F}. \tag{24}$$

Eq. 24 indicates that the similarity between two experts is jointly determined by three factors: the spectral weights $\sigma_{c,i}$, the overlap of left singular directions, and the overlap of right singular directions. Therefore, different optimization dynamics that modify the spectral distribution of G_c can directly affect the final similarity between experts.

Spectral dynamics under direct optimization. In direct optimization, the trainable variable is the dense transformation matrix G_c. Let

$$\Gamma_c = \nabla_{G_c} \mathcal{L}_c \tag{25}$$

denote the gradient of the loss with respect to G_c.

The continuous-time approximation of gradient descent is given by

$$\dot{G}_c = -\Gamma_c. \tag{26}$$

At each time step, G_c admits an instantaneous SVD

$$G_c(t) = M_c(t) \Sigma_c(t) N_c(t)^\top. \tag{27}$$

Taking the derivative with respect to time gives

$$\dot{G}_c = \dot{M}_c \Sigma_c N_c^\top + M_c \dot{\Sigma}_c N_c^\top + M_c \Sigma_c \dot{N}_c^\top. \tag{28}$$

Multiplying the above equation by $M_c^\top$ and N_c gives

$$M_c^\top \dot{G}_c N_c = M_c^\top \dot{M}_c \Sigma_c + \dot{\Sigma}_c + \Sigma_c \dot{N}_c^\top N_c. \tag{29}$$

Because M_c and N_c are orthogonal matrices, $M_c^\top \dot{M}_c$ and $\dot{N}_c^\top N_c$ are skew-symmetric matrices. Therefore, their diagonal elements are zero, leading to

$$(M_c^\top \dot{G}_c N_c)_{ii} = \dot{\sigma}_{c,i}. \tag{30}$$

Substituting Eq. 26, we obtain

$$\dot{\sigma}_{c,i} = -m_{c,i}^\top \Gamma_c n_{c,i}, \tag{31}$$

where $m_{c,i}$ and $n_{c,i}$ denote the i-th columns of M_c and N_c, respectively.

Eq. 31 shows that under direct optimization, the evolution of each singular value is determined only by the projection of the gradient onto the corresponding singular direction. In particular, the update magnitude has no explicit dependence on the current singular value $\sigma_{c,i}$. Therefore, different spectral components can receive comparable gradient updates as long as their gradient projections are similar.

Spectral dynamics under SVD-based optimization. Different from direct optimization, SVD-based optimization directly treats the factors M_c, Σ_c, and N_c as optimization variables:

$$G_c = M_c \Sigma_c N_c^\top. \tag{32}$$

The differential of G_c is

$$dG_c = dM_c \Sigma_c N_c^\top + M_c d\Sigma_c N_c^\top + M_c \Sigma_c dN_c^\top. \tag{33}$$

Using the chain rule, the gradients of the three factors are

$$\nabla_{M_c} \mathcal{L}_c = \Gamma_c N_c \Sigma_c, \tag{34}$$

$$\nabla_{N_c} \mathcal{L}_c = \Gamma_c^\top M_c \Sigma_c, \tag{35}$$

and

$$\dot{\Sigma}_c = -\operatorname{diag}(M_c^\top \Gamma_c N_c). \tag{36}$$

The induced matrix dynamics becomes

$$\dot{G}_c = \dot{M}_c \Sigma_c N_c^\top + M_c \dot{\Sigma}_c N_c^\top + M_c \Sigma_c \dot{N}_c^\top. \tag{37}$$

Substituting the factor gradients yields

$$\begin{aligned} \dot{G}_c = &- \Gamma_c N_c \Sigma_c^2 N_c^\top \\ &- M_c \operatorname{diag}(M_c^\top \Gamma_c N_c) N_c^\top \\ &- M_c \Sigma_c^2 M_c^\top \Gamma_c. \end{aligned} \tag{38}$$

Let

$$\widetilde{\Gamma}_c = M_c^\top \Gamma_c N_c. \tag{39}$$

Projecting Eq. 38 onto the current singular coordinate system gives

$$M_c^\top \dot{G}_c N_c = -\left(\widetilde{\Gamma}_c \Sigma_c^2 + \operatorname{diag}(\widetilde{\Gamma}_c) + \Sigma_c^2 \widetilde{\Gamma}_c \right). \tag{40}$$

For each element,

$$(M_c^\top \dot{G}_c N_c)_{ij} = -(\sigma_{c,i}^2 + \sigma_{c,j}^2 + \delta_{ij})(\widetilde{\Gamma}_c)_{ij}. \tag{41}$$

Under the instantaneous SVD assumption, the diagonal entries correspond to singular value evolution:

$$\dot{\sigma}_{c,i} = -(2\sigma_{c,i}^2 + 1) m_{c,i}^\top \Gamma_c n_{c,i}. \tag{42}$$

Compared with Eq. 31, the spectral update in SVD-based optimization is explicitly modulated by the current singular values. Large singular-value directions receive amplified updates, while small singular-value directions receive weaker updates. Therefore, SVD-based optimization introduces a spectral preference mechanism, which encourages the concentration of adaptation energy into a small number of dominant spectral components.

Effect of spectral concentration on expert similarity. Due to the spectral-dependent dynamics in Eq. 42, SVD-based optimization tends to produce a more concentrated singular-value distribution:

$$\sigma_{c,k_1} \gg \sigma_{c,k_2}, \ldots, \sigma_{c,k_r}, \tag{43}$$

where k_1 denotes the dominant spectral component.

Consequently, the expert matrix can be approximated by its dominant spectral component:

$$G_c \approx \sigma_{c,k_c} m_{c,k_c} n_{c,k_c}^\top. \tag{44}$$

Substituting Eq. 44 into Eq. 24, the similarity between two experts becomes

$$\rho_{ij} \approx (m_{i,k_i}^\top m_{j,k_j})(n_{i,k_i}^\top n_{j,k_j}). \tag{45}$$

Therefore, after spectral concentration, expert similarity is mainly determined by the overlap between their dominant left and right spectral directions.

For different categories, the corresponding gradients Γ_i and Γ_j are generally different. According to Eq. 42, the dominant spectral components are determined by the gradient projections:

$$m_{c,k}^\top \Gamma_c n_{c,k}. \tag{46}$$

Thus, different categories tend to select different dominant spectral directions. Since these directions are located in a high-dimensional parameter space, their overlaps are likely to be small:

$$m_{i,k_i}^\top m_{j,k_j} \approx 0, \qquad n_{i,k_i}^\top n_{j,k_j} \approx 0. \tag{47}$$

Combining Eq. 45, the expert similarity is consequently reduced:

$$\rho_{ij}^{\mathrm{SVD}} < \rho_{ij}^{\mathrm{Direct}}. \tag{48}$$

Therefore, the essential effect of SVD-based parameterization is not changing the representational capacity of the LoRA transformation, but modifying the optimization trajectory through spectral-dependent gradient dynamics. This mechanism promotes concentrated spectral structures and makes different category experts rely on more category-specific dominant spectral directions, thereby reducing their similarity.

A.4 Additional Benchmark Details

ODinW-13 benchmark. Following ZiRa Deng et al. (2024), we evaluate sequential specialization on the Object Detection in the Wild benchmark Li et al. (2022) and assess the preservation of zero-shot generalization on COCO Lin et al. (2014). ODinW-13 consists of 13 heterogeneous object detection datasets: Aerial Maritime Drone (Ae), Aquarium (Aq), Cottontail Rabbits (Co), EgoHands (Eg), Mushrooms (Mu), Packages (Pa), PascalVOC (Pv), Pistols (Pi), Pothole (Po), Raccoon (Ra), Shellfish (Sh), Thermal(Th), and Vehicles (Ve). These datasets cover diverse object categories, visual environments and imaging conditions, providing a challenging setting for evaluating sequential adaptation across heterogeneous domains. To further evaluate open-vocabulary generalization, we use ZCOCO to denote zero-shot performance on the COCO validation set, a widely adopted benchmark in VLOD and OVOD that is also referred to as OV-COCO. The validation set contains 5,000 images annotated with 80 common object categories, such as person, car, and dog. After sequentially learning all ODinW-13 tasks, the model is evaluated on each learned task and MS COCO without additional fine-tuning. We use Avg to denote the average mAP across all learned tasks. Their harmonic mean H measures the balance between continual adaptation and open-vocabulary generalization.

ODinW-O benchmark. Following DitHub Cappellino et al. (2026), we additionally evaluate our method on ODinW-O, an overlapping-category benchmark derived from ODinW-35 Li et al. (2022). ODinW-O is designed to evaluate incremental adaptation when semantic categories recur across tasks under different domains. For example, the category car appears in aerial, street-level, and autonomous-driving scenes. The benchmark retains only categories shared by at least two tasks and removes tasks without recurring categories. The datasets and their overlapping categories in ODinW-O are as follows:

Table 7: Parameter efficiency comparison on ODinW-13. $\mathcal{E}$ denotes encoder adaptation.

Method	Params	ZCOCO	Avg	H
G-DINO	667M	47.4	46.8	47.1
G-DINO$^{\mathcal{E}}$	92M	38.6	54.7	45.3
ZiRa	18M	46.3	58.0	51.5
DitHub ($r = 16$)	147M	47.0	62.2	53.5
CADT ($r = 1$)	0.5M	46.7	52.0	49.2
CADT ($r = 2$)	1.1M	47.3	53.4	50.2
CADT ($r = 4$)	2.7M	**47.5**	55.6	51.2
CADT ($r = 8$)	7.2M	47.3	59.1	52.5
CADT ($r = 16$)	21.6M	47.0	62.5	**53.7**
CADT ($r = 32$)	71.8M	46.8	**62.8**	53.6

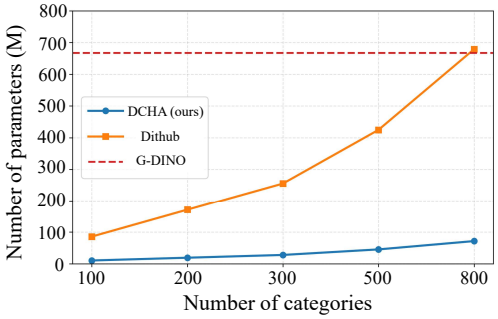


Figure 6: Parameter growth with the number of categories.

- **Aerial Maritime Drone**: boat and car;
- **Pascal VOC**: boat, car, dog, and person;
- **Thermal**: dog and person;
- **Vehicles**: car and truck;
- **Hard Hat Workers**: person;
- **Self-Driving Cars**: car and truck.

Apart from the task composition, all evaluation metrics follow the same definitions as those used for ODinW-13.

A.5 Additional Experiments

Parameter efficiency and scalability. Table 7 evaluates the effect of LoRA rank and compares the parameter efficiency of different methods. Increasing the rank from 1 to 16 consistently improves Avg from 52.0 to 62.5 and H from 49.2 to 53.7, while keeping ZCOCO stable. Notably, CADT with $r = 8$ outperforms ZiRa on all three metrics using only 7.2M parameters compared with 18M. With $r = 16$, CADT achieves the highest H of 53.7, exceeding DitHub in Avg and H by 0.3 and 0.2 points while reducing the parameter count from 147M to 21.6M by 85.3%. Further increasing the rank to 32 provides only a marginal Avg improvement and slightly reduces H despite the substantial parameter increase. We therefore adopt $r = 16$ for the best balance between performance and parameter efficiency. Figure 6 further compares parameter growth as the number of categories increases. The storage cost of DitHub grows rapidly and exceeds the 667M parameters of the original G-DINO model at 800 categories, whereas CADT remains below 100M, demonstrating substantially better scalability for category-level adaptation.

Robustness to task order. We compare DitHub and CADT under three task orders with different initial tasks in Table 8. CADT consistently outperforms DitHub across all orders. On average, CADT achieves an Avg of 62.0 and an H of 53.4, compared with 61.7 and 53.3 for DitHub. The consistently small performance variation across task orders demonstrates that CADT is robust to the choice of the initial task.

Table 8: Performance comparison between DitHub and CADT under different task orders.

Shots	First task	DitHub			CADT		
		ZCOCO	Avg	H	ZCOCO	Avg	H
Full	Th	46.8	61.7	53.2	46.9	61.9	53.4
	Sh	46.8	61.3	53.1	46.8	61.5	53.4
	Pa	47.0	62.2	53.5	47.0	62.5	53.7
Mean		**46.9**	61.7	53.3	**46.9**	**62.0**	**53.4**

Effect of category-specific initialization in DCT. We compare identical, random, and orthogonal initialization strategies in Table 9. Identical initialization places all category transformations at the same directional starting point, which may encourage optimization toward similar solutions. Random initialization introduces category-specific

Table 9: Ablation of the initialization.

Initialization	ZCOCO	Avg	H
Identical	46.8	61.3	53.1
Random	**47.0**	61.6	53.3
Orthogonal	**47.0**	**62.5**	**53.7**

Table 13: Effects of auxiliary loss weights.

λ_{d}	0.01	0.1	0.3	0.5	0.7
ZCOCO	47.0	47.0	**47.1**	46.9	47.2
Avg	61.6	**62.5**	61.3	61.1	60.9
H	53.3	**53.7**	53.3	53.1	53.2

λ_{s}	0.01	0.1	0.3	0.5	0.7
ZCOCO	**47.0**	**47.0**	46.9	47.0	46.8
Avg	61.6	**62.5**	61.3	61.1	60.9
H	53.3	**53.7**	53.1	53.1	52.9

λ_{h}	0.01	0.1	0.3	0.5	0.7
ZCOCO	**47.1**	47.0	47.0	47.0	47.0
Avg	60.6	**62.5**	61.6	61.8	61.6
H	53.0	**53.7**	53.3	53.4	53.3

diversity and improves Avg and H by 0.3 and 0.2 points, respectively, but lacks explicit control over directional redundancy. In contrast, orthogonal initialization provides distinct and non-redundant directional bases while preserving the initial effective update, further improving Avg and H to 62.5 and 53.7, demonstrating the benefit of structured directional initialization for category-specific adaptation.

Choice of descriptor construction in DCT. We compare four strategies for constructing the DCT's descriptor in Table 10. Max uses the largest coefficient along the pooled dimension of each LoRA component, which may make the descriptor sensitive to isolated large values rather than the overall structure of the LoRA components. Abs constructs the descriptor from the absolute values of coefficients in the LoRA components, discarding their signs. Although it achieves the highest ZCOCO of 47.1, its Avg and H are 3.6 and 1.4 points lower than those of Mean. Log Product computes each rank i entry as $[f_w]_i = \log(1 + \|(A_w)_{i,:}\|_2 \|(B_w)_{:,i}\|_2)$, where $\|\cdot\|_2$ denotes the Euclidean norm. This captures the magnitude of the corresponding rank-one LoRA update, but the norm-based product discards signed information and the logarithm compresses differences between rank-wise entries. Mean aggregates the signed statistics of LoRA and achieves the highest Avg of 62.5 and H of 53.7. Therefore, we adopt Mean for descriptor construction in DCT.

Table 10: Ablation study on descriptor construction in DCT.

Strategy	ZCOCO	Avg	H
Max	46.9	60.1	52.7
Mean	47.0	**62.5**	**53.7**
Abs	**47.1**	58.9	52.3
Log Product	46.6	57.8	51.6

Effect of merging coefficient in CAC. We evaluate the fixed merging coefficient α in Table 11. As α increases from 0.3 to 0.9, Avg rises from 59.7 to 62.5 and H rises from 52.4 to 53.7. ZCOCO fluctuates slightly over this range but also reaches its highest value of 47.0 at $\alpha = 0.9$. Further increasing α to 0.95 reduces ZCOCO, Avg, and H to 46.8, 62.0, and 53.3, respectively, suggesting that placing too much weight on the current directional composition weakens the benefit of incorporating historical information. Thus, we adopt $\alpha = 0.9$ as the merging coefficient.

Table 11: Effect of the merging coefficient α in CAC.

α	0.3	0.5	0.7	0.9	0.95
ZCOCO	46.7	46.8	46.6	**47.0**	46.8
Avg	59.7	60.2	61.4	**62.5**	62.0
H	52.4	52.7	53.0	**53.7**	53.3

Effect of freezing the amplitude matrix in CAC. We investigate whether S_c should remain trainable during CAC in Table 12. Updating S_c allows the current task to alter the direction-wise strengths learned from previous tasks, which may introduce amplitude fluctuations and increase category adaptation drift. Freezing S_c improves H from 52.8 to 53.7 and reduces $\bar{F}$ from 1.1 to 0.1, while also decreasing Δ_{ZS} from 0.5 to 0.4. These results show that preserving the historical amplitudes while consolidating M_c and N_c provides more stable adaptation for recurring categories.

Table 12: Ablation study on freezing the amplitude matrix S_c in CAC.

S_c setting	H	$\Delta_{\mathrm{ZS}}\downarrow$	$\bar{F}\downarrow$
Unfrozen	52.8	0.5	1.1
Frozen	**53.7**	**0.4**	**0.1**

Effectiveness of auxiliary losses. We study the effects of λ_{d}, λ_{s}, and λ_{h} in Table 13. For λ_{d}, Avg and H peak at 62.5 and 53.7 when the weight is 0.1, and larger weights reduce both metrics. The same trend is observed for λ_{s}, with Avg and H decreasing to 60.9 and 52.9 at 0.7. For λ_{h}, increasing the weight from 0.01 to 0.1 improves Avg by 1.9 points and H by 0.7 points, while further increases yield lower performance. ZCOCO remains stable across the tested weights. These results show that moderate loss weights provide the best balance between adaptation and regularization. Therefore, we set $\lambda_{\mathrm{d}} = \lambda_{\mathrm{s}} = \lambda_{\mathrm{h}} = 0.1$.

Table 14: Effect of the transition position.

Transition Point	10%	30%	50%	70%	90%
ZCOCO	46.4	46.7	46.8	**47.0**	46.8
Avg	60.3	61.3	61.0	**62.5**	60.0
H	52.4	53.0	53.0	**53.7**	52.6

Table 15: Effect of category organization granularity.

Strategy	Method	Params	ZCOCO	Avg	H
3 groups	DitHub	17.5M	47.0	48.7	47.8
	CADT	7.9M	**47.2**	**50.3**	**48.7**
category-level	DitHub	147.0M	**47.0**	62.2	53.5
	CADT	21.6M	**47.0**	**62.5**	**53.7**

Transition position. We evaluate the transition position between shared-anchor learning and category-specific adaptation in Table 14. An early transition at 10% yields an Avg of 60.3 and an H of 52.4, suggesting that the shared anchor has not yet captured sufficient transferable knowledge. Moving the transition to 70% improves ZCOCO, Avg, and H to their highest values of 47.0, 62.5, and 53.7, respectively. A later transition at 90% reduces Avg and H to 60.0 and 52.6, as insufficient iterations remain for category-specific adaptation. Therefore, we set the transition position to 70% to balance shared-anchor learning and category-specific optimization.

Class grouping. To further investigate the effect of category granularity, we explore two strategies for sharing knowledge among related categories: WordNet-based grouping and CLIP-based clustering. For the WordNet-based grouping, it uses the WordNet hierarchy to manually organize all categories into three coarse semantic groups: *animals*, *vehicles*, and *objects*. The resulting groups are as follows:

- **Animals:** fish, jellyfish, penguin, puffin, shark, starfish, stingray, cottontail-rabbit, raccoon, crab, lobster, shrimp, dog, cat, cow, sheep, horse, bird, elephant, bear, zebra, giraffe, chanterelle
- **Vehicles:** boat, car, jetski, aeroplane, airplane, bicycle, bus, motorbike, motorcycle, train, truck, ambulance, skateboard, surfboard, skis, snowboard
- **Objects:** dock, lift, hand, person, package, pistol, pothole, bottle, chair, diningtable, pottedplant, sofa, tvmonitor, traffic light, fire hydrant, stop sign, parking meter, bench, backpack, umbrella, handbag, tie, suitcase, frisbee, sports ball, kite, baseball bat, baseball glove, tennis racket, wine glass, cup, fork, knife, spoon, bowl, banana, apple, sandwich, orange, broccoli, carrot, hot dog, pizza, donut, cake, bed, toilet, laptop, mouse, remote, keyboard, cell phone, microwave, oven, toaster, sink, refrigerator, book, clock, vase, scissors, teddy bear, hair drier, toothbrush

We compare the category organization strategies of DitHub and CADT in Table 15. Under both grouping granularities, CADT consistently outperforms DitHub while requiring substantially fewer parameters. With three groups, CADT improves ZCOCO, Avg, and H by 0.2, 1.6, and 0.9 points, respectively, while reducing the parameter count from 17.5M to 7.9M. At the category level, CADT achieves the same ZCOCO as DitHub and improves Avg and H by 0.3 and 0.2 points, while reducing the parameter count from 147.0M to 21.6M by 85.3%. These results show that the compact category-specific transformations of CADT enable fine-grained adaptation without the large storage overhead of complete category-specific LoRA modules.

A.6 Metrics for Category Adaptation Analysis

We employ complementary metrics to analyze the inter-category diversity and intra-category stability of category-specific adaptations. Unless otherwise specified, all metrics are computed on the effective category-specific update at each encoder layer,

$$\Delta W_c^l = B_w^l G_c^l A_w^l, \tag{49}$$

where l denotes the encoder layer and c denotes the category. The resulting metric values are averaged across encoder layers.

A.6.1 Inter-Category Diversity Metrics

We first evaluate whether different categories learn sufficiently diverse adaptation updates. Given two category-specific update matrices $X, Y \in \mathbb{R}^{n\times m}$, we center them as

$$\widetilde{X} = HX, \qquad \widetilde{Y} = HY, \qquad H = I - \frac{1}{n}\mathbf{1}\mathbf{1}^\top, \tag{50}$$

where $\widetilde{X}$ and $\widetilde{Y}$ denote the centered updates, $H \in \mathbb{R}^{n\times n}$ is the centering matrix, I is the identity matrix, $\mathbf{1} \in \mathbb{R}^n$ is the all-ones vector, and n denotes the number of rows in each update matrix. Their linear CKA similarity is computed as

$$\mathrm{CKA}(X,Y) = \frac{\left\|\widetilde{X}^\top \widetilde{Y}\right\|_F^2}{\left\|\widetilde{X}^\top \widetilde{X}\right\|_F \left\|\widetilde{Y}^\top \widetilde{Y}\right\|_F}, \tag{51}$$

where $\|\cdot\|_F$ denotes the Frobenius norm and $\top$ denotes matrix transpose. For category-similarity analysis, we report the squared CKA score

$$\mathcal{S}(X,Y) = \mathrm{CKA}(X,Y)^2, \tag{52}$$

where $\mathcal{S}$ denotes the pairwise similarity between two category-specific updates. Since category updates generally exhibit relatively high CKA similarities, we square the CKA scores to enhance the contrast among highly similar updates and facilitate visualization. A larger $\mathcal{S}$ indicates stronger structural similarity and thus a higher degree of update homogenization. Accordingly, the pairwise category discrepancy is defined as

$$\mathcal{D}(X,Y) = 1 - \mathrm{CKA}(X,Y)^2, \tag{53}$$

where $\mathcal{D}$ denotes the discrepancy between two category-specific updates. A larger value indicates more distinct category-specific adaptations, whereas a smaller value indicates more homogeneous updates.

A.6.2 Intra-Category Stability Metrics

We further evaluate the stability of the same category across sequential tasks using category consistency and representation drift.

Category consistency. We measure category consistency using linear CKA between two category-specific updates. For the visualization in Fig. 5, the adaptation learned at the first task is fixed as the reference, and the consistency of category c after task t is defined as

$$\mathrm{Cons}_c^t = \mathrm{CKA}\left(\Delta W_c^t, \Delta W_c^1\right), \tag{54}$$

where ΔW_c^1 denotes the category-specific update obtained after the first task, ΔW_c^t denotes its update after task t, and Cons_c^t denotes the corresponding cross-task category consistency. Unlike the inter-category similarity metric, we directly use the original CKA score without squaring. A higher value indicates that the adaptation structure of category c remains more consistent with its initial state throughout sequential learning.

Representation drift. We further quantify the accumulated parameter deviation using the relative Frobenius distance. Using the first-task adaptation as the reference, the drift of category c after task t is defined as

$$\mathcal{R}_c^t = \frac{\left\|\Delta W_c^t - \Delta W_c^1\right\|_F}{\|\Delta W_c^1\|_F + \epsilon}, \tag{55}$$

where $\mathcal{R}_c^t$ denotes the relative representation drift, $\|\cdot\|_F$ denotes the Frobenius norm, and ϵ is a small constant for numerical stability. A smaller $\mathcal{R}_c^t$ indicates less deviation from the initially learned category adaptation and therefore better preservation of category-specific knowledge.