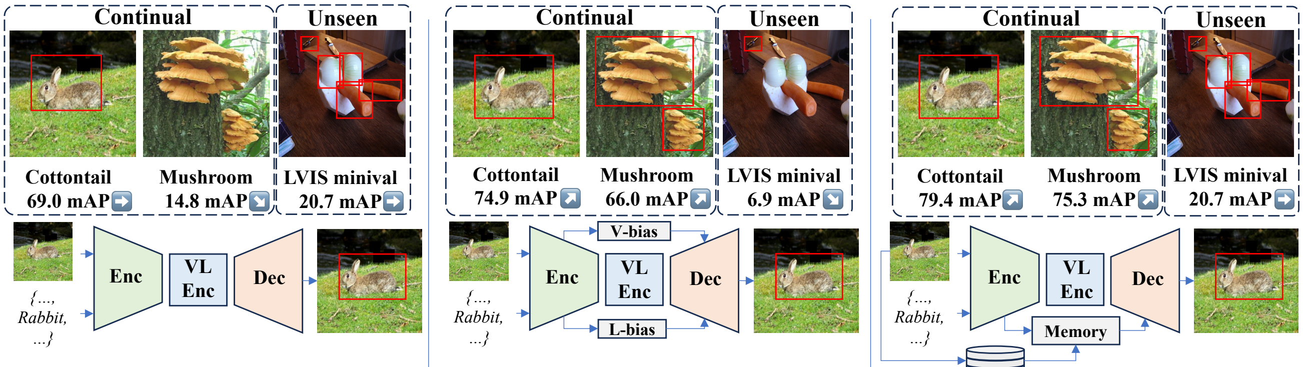
MR-GDINO: Efficient Open-World Continual Object Detection

Bowen Dong[1,2] Zitong Huang[1] Guanglei Yang[1] Lei Zhang[2] Wangmeng Zuo[1]
[1]Harbin Institute of Technology [2]The Hong Kong Polytechnic University
{cndongsky, zitonghuang99}@gmail.com cslzhang@comp.polyu.edu.hk wmzuo@hit.edu.cn



(a) Zero-shot Grounding DINO **(b) Continual Object Detection** **(c) Open-world Continual Detection (Ours)**

Figure 1. (a) Pretrained open-world (OW) detectors [34] show strong generalization abilities on unseen data but cannot benefit from few-shot annotations. (b) Continual detectors [8] built on OW detectors with continual learning show improved mAP on seen data but suffer from forgetting for unseen objects. (c) Our OW continual detector MR-GDINO via memory and retrieval improves detection abilities on seen classes while preserving OW abilities on unseen classes.

Abstract

Open-world (OW) recognition and detection models show strong zero- and few-shot adaptation abilities, inspiring their use as initializations in continual learning methods to improve performance. Despite promising results on seen classes, such OW abilities on unseen classes are largely degenerated due to catastrophic forgetting. To tackle this challenge, we propose an open-world continual object detection task, requiring detectors to generalize to old, new, and unseen categories in continual learning scenarios. Based on this task, we present a challenging yet practical OW-COD benchmark to assess detection abilities. The goal is to motivate OW detectors to simultaneously ***preserve*** *learned classes,* ***adapt*** *to new classes, and* ***maintain*** *open-world capabilities under few-shot adaptations. To mitigate forgetting in unseen categories, we propose MR-GDINO, a strong, efficient and scalable baseline via* ***memory and retrieval*** *mechanisms within a highly scalable memory pool. Experimental results show that existing continual detectors suffer from severe forgetting for both seen and unseen categories. In contrast, MR-GDINO largely mitigates forgetting with only 0.1% activated extra parameters, achieving state-of-the-art performance for old, new, and unseen categories.*

1. Introduction

Open-world (OW) recognition and detection models [2, 41, 50, 57, 62, 63] has shown remarkable progress in effectively recognize and localize generalized objects [17, 32, 52, 61] with different granularity [28, 54]. By learning with vast semantic-rich data [3, 25, 26, 32, 41, 44, 52, 60], even without supervised by any bounding boxes from unseen classes, OW models (*e.g.*, object detection networks) can generalize well under a *zero-shot learning paradigm in open-world scenarios* (*e.g.*, zero-shot Grounding DINO for OW in Fig 1(a)). Benefited from highly generalized feature representation of detectors, such detectors can be also adapt to new classes via many- or few-shot fine-tuning [7, 18, 21, 29, 34], thus performing better on target classes.

By sequentially repeating the fine-tuning procedure, one can formulate the updated OW detectors as a continual learning paradigm [55, 65]. This formulation is intuitive and more practical than traditional open-world learning [30] or continual learning [65] by two-fold. First, pretrained open-world (OW) detectors offer a robust initial representation that supports both zero-shot learning and rapid adaptation. Second, OW detectors are designed to encounter various out-of-distribution scenarios, but still suffer from performance drops by shifts in domain [5, 39, 47] and previously unseen categories [16, 23, 49, 51]. Those models with fast adaptation can effectively address this issue, enhancing performance under real-world deployment conditions. Thus, we anticipate that OW detectors will preserve the advantages of open-world learning and demonstrate strong generalization across both known and novel categories.

As shown in Fig 1(b), prior studies [8, 29, 30, 34] on OW detectors [34] in continual learning make feature representations strictly adapt to target classes domain [9, 18], and break original well-aligned visual-language representation. Though obtaining better performance than zero-shot OW detectors among seen categories, traditional continual learning frameworks for OW detectors still suffer degree of catastrophic forgetting for previously seen categories. Moreover, after continual adaptations on seen categories, the unseen categories detection capabilities of obtained detectors degenerate. This limitation constrains the applicability of OW detectors in real-world scenarios. To tackle this challenge, our research pursues two objectives, including: 1) assessing catastrophic forgetting in OW detectors across various learning frameworks, and 2) developing continual learning strategies specifically tailored for OW detectors for promising detection abilities on seen and unseen categories.

To this end, we propose the *open-world continual object detection task*, which requires optimized open-world detectors to simultaneously **preserve** knowledge of old classes, **adapt** to new classes, and **maintain** detection capabilities for unseen classes under continual few-shot adaptations. Due to the lack of proper evaluation toolkits, building upon this task, we propose a challenging yet practical benchmark OW-COD, specifically designed to evaluate anti-catastrophic forgetting capabilities across old, new, and unseen categories within continual learning frameworks for OW detectors. Specifically, OW-COD includes two groups of data. The former is few-shot training data with corresponding evaluation samples from various domains [29], which are sequentially utilized to optimize OW detectors via continual learning paradigm, and evaluate the detection performance for both old and new seen classes under class-incremental settings. And the latter is large-scale open-world object detection evaluation data [17], which are used to assess the detection accuracy of unseen categories against catastrophic forgetting. The combination of evaluation for both seen and unseen categories fits the goal of our task, and provides a comprehensive benchmark for continual learning frameworks under open-world detection scenarios.

Based on OW-COD benchmark, we construct a strong baseline method to achieve the goal of our task. Based on prior studies [13, 29], we argue that explicit visual-language interaction module is the key component for open-world detection. To enhance the anti-catastrophic forgetting capability of these modules for better unseen categories detection ability, we propose a strong baseline MR-GDINO for OW-COD benchmark, a highly scalable open-world continual object detection method via ***M**emory* and ***R**etrieval* mechanisms. Specifically, MR-GDINO employs a scalable memory pool, which efficiently caches parameter triplets regrading new concepts and visual-language interactions from continual learning steps. And during inference, MR-GDINO enables to adaptively retrieve optimal parameter triplet to detect objects in previously learned, newly adapted, or open-world scenarios. The memory and retrieval mechanisms ensures the flexibility, scalability, and performance of MR-GDINO, thus preserving detection capabilities of old, new and unseen open-world categories.

Extensive experiments are conducted on our proposed OW-COD between different continual object detection frameworks and MR-GDINO. As shown in Fig 1(c), with only tiny activated additional parameters our MR-GDINO largely surpasses GDINO on seen classes with only few-shot continual adaptations. Moreover, owning to robust retrieval machanism, MR-GDINO enables simultaneous promising performance between unseen and seen classes.

In summary, our contributions are shown as follows:

- We present OW-COD, a challenging yet practical benchmark to assess seen and unseen classes detection abilities of OW detectors under few-shot continual adaptations.
- We propose MR-GDINO, a *strong, efficient, and scalable* OW continual detector via *memory and retrieval mechanisms* with a highly scalable memory pool.
- By *only 0.1% activated extra parameters*, MR-GDINO effectively improves detection capabilities for continually seen categories under few-shot adaptation, meanwhile ensuring open-world detection ability without forgetting.

2. Related Work

2.1. Open-World Object Detection

Open-world (OW) object detection [2, 29, 40, 50, 63] aims to develop optimal detectors capable of recognizing both seen and unseen categories in real-world scenarios by vast semantic-rich multi-modal data [3, 26, 41, 44, 52, 56, 60]. A crucial component in OW detector design is the visual-language (VL) interaction module [29, 34, 50], which links visual features with text embeddings, influencing detection capabilities. OW detectors are broadly classified into matching-based detectors [56, 58] which use pretrained text embeddings to identify localized objects, and fusion-based detectors [29, 34, 50, 63] which incorporate attention modules [29, 34] or ranking gates [50] to merge visual and language features for accurate classification. However, seldom studies [8] explore catastrophic forgetting in OW detectors under continual adaptations. In contrast, OW-COD investigates this issue in continual adaptations, and MR-GDINO ensures promising abilities on both seen and unseen classes.

Few-shot object detection with OW detectors. Our work shares similarities with few-shot object detection. Pretrained OW detectors [34, 53] can adapt rapidly to target domains using few-shot training samples [11, 22, 53] for better performance. However, this often results in poor generalization to unseen categories [8]. In contrast, MR-GDINO demonstrates robust performance on both seen and unseen categories during continual few-shot adaptations.

2.2. Continual Object Detection

Continual object detection (COD) [12, 35, 64] aims to learn detectors that incorporate new classes while retaining knowledge of prior ones. Early methods like ILOD[45] use pseudo-label distillation to address catastrophic forgetting [31, 42], with recent works improving architectures and training strategies [14, 35, 46, 64]. However, few studies [8] focus on OW-COD. In contrast, our MR-GDINO introduces a retrieval-based [9, 55] approach for continual few-shot adaptations with pretrained OW models [34, 50], preventing forgetting and extending COD to practical scenarios.

3. Open-World Continual Object Detection

3.1. Task Definition

Building upon COD and OWOD, we formulate the open-world continual object detection task. Given an open-world (OW) object detector f pretrained on a large-scale dataset $\mathbb{D}_{\text{pre}}$, as well as a sequence of training set $\{\mathbb{D}_1, \ldots, \mathbb{D}_T\}$ with size of T, we aim to optimize f with corresponding parameters θ_f by sequentially learning on each $\mathbb{D}_i$. Such that the optimized $f(\cdot;\theta_f)$ enables to accurately detect both previously learned old classes $\mathbb{C}_1 \cup \cdots \cup \mathbb{C}_{T\text{-}1}$ and newly learned classes $\mathbb{C}_T$, where $\mathbb{C}_i$ represents the label set of corresponding $\mathbb{D}_i$. Meanwhile, $f(\cdot;\theta_f)$ should be generalized well on open-world evaluation dataset $\mathbb{D}^{\text{val}}_{\text{unseen}}$ with corresponding large-scale and diverse label space $\mathbb{C}_{\text{unseen}}$. The goal of our proposed task is to motivate OW detectors to simultaneously **preserve** learned classes, **adapt** to new classes, and **maintain** open-world capabilities, which is critical for OW detectors to simultaneously adapt to varying new environment and keep generalization abilities.

3.2. Benchmark Construction

After defining the task, we formulate corresponding OW-COD dataset as the data source for continual learning and universal evaluation for old, new, and unseen classes. Generally, OW-COD is collected from existing object detection datasets [17, 29] and broadly divided to two groups, *i.e.*, seen category data and unseen category data. For seen category data, we leverage 13 subsets (from "Aerial" to "Vehicle") from ODinW-13 [29], and assign $\{\mathbb{D}_1, \ldots, \mathbb{D}_T\}$ by ascending dictionary order of subsets. The label space among $\{\mathbb{D}_1, \ldots, \mathbb{D}_T\}$ are usually non-overlapped and fit the requirements of our task. During training of each step t, only images from $\mathbb{D}_t$ are visible. Notably, to simulate practical fast adaptation scenarios and increase the challenge of the benchmark, a few-shot training setting is adopted in OW-COD. This setting requires continual OW detectors to effectively mitigate the impact of both overfitting and catastrophic forgetting, thereby enabling robust detection abilities for old, new, and unseen categories. And for the unseen category data, to better align with real-world deployment scenarios, LVIS [17] *minival* set with $\sim$5k validation images and 1,203 categories is leveraged to empirically assess detection performance for unseen classes. This subset is only used for evaluation. Leveraging the dataset's large-scale and highly diverse label space facilitates empirical analysis of anti-forgetting abilities for unseen classes under continual adaptation. Statistics of the MR-GDINO training and evaluation data are shown in the suppl.

3.3. Metrics of OW-COD

Average Precision. Following the work on continual object detection [8, 12, 35, 64] and open-world object detection [2, 29, 40, 50, 63], the mean average precision (mAP) is reported for each subset to quantitatively assess the performance of learned open-world (OW) detectors under the continual learning paradigm. Specifically, per-subset average precision (AP) is provided to evaluate the detection performance of continual OW detectors after few-shot continual adaptations. Additionally, the mean AP for previously learned, newly seen, and unseen categories is reported to summarize the overall performance.

Average rank. Inspired by previous benchmarks [29, 61], OW-COD also incorporates average rank as auxiliary metric to measure the relative performance of existing continual OW detectors. Specifically, OW-COD first ranks all models within each subset. For the K subsets of seen categories, let R^i_j denote the rank of the i-th subset by the j-th detector. The average rank of seen categories R^{seen}_j is then defined as:

$$R^{\text{seen}}_j = \frac{\sum_{i=1}^{K} R^i_j}{K}. \tag{1}$$

Similarly, we define the unseen classes average rank R^{unseen}_j for j-th detector. Finally, overall rank R^{avg}_j is calculated by:

$$R^{\text{avg}}_j = \sqrt{\frac{{R^{\text{seen}}_j}^2 + {R^{\text{unseen}}_j}^2}{2}}. \tag{2}$$

The merit of this ranking lies in the fact that a detector can achieve a higher rank only if it performs well on both seen and unseen categories, thus emphasizing its ability to mitigate catastrophic forgetting for both old and new classes.

3.4. Relation with Counterparts

Comparison with COD. COD [8, 12, 35, 64] typically split annotations from entire datasets (*e.g.*, COCO [32]) by dividing label sets into groups, which is not practical since novel categories often appear in unseen scenarios [49, 51],

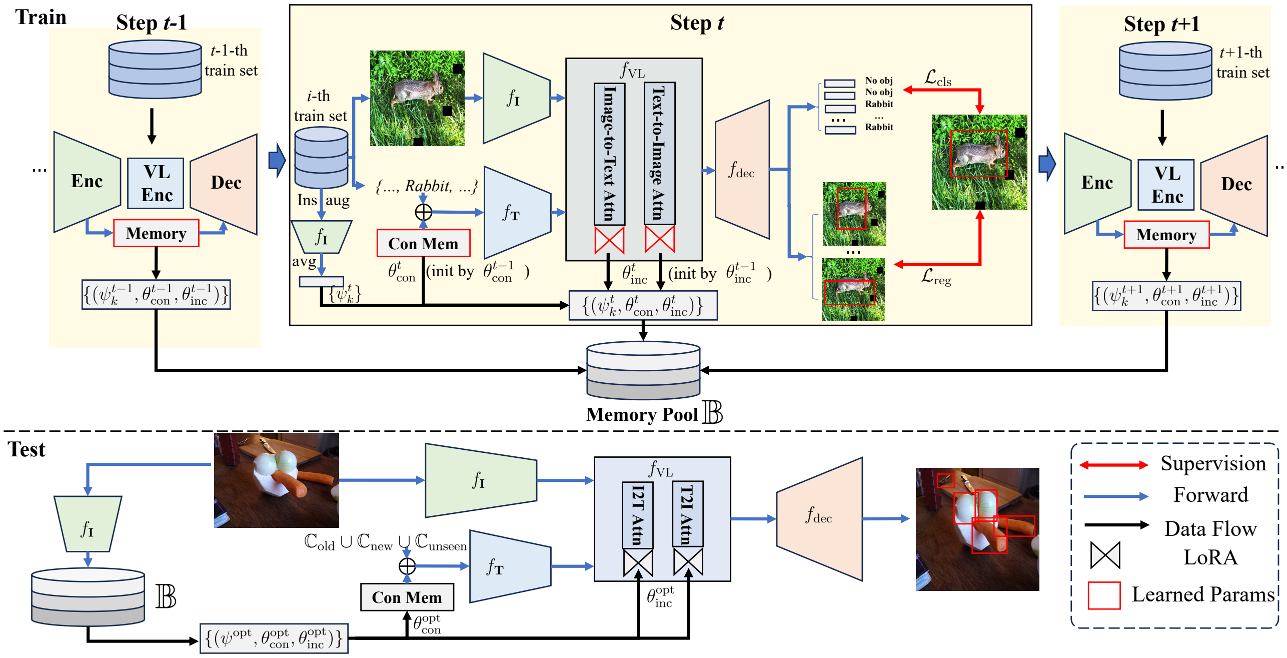

Train
Step t-1
t-1-th train set
Enc
VL Enc
Dec
Memory
$\{(\psi_k^{t-1}, \theta_{\text{con}}^{t-1}, \theta_{\text{inc}}^{t-1})\}$
Step t
i-th train set
Ins aug
$f_{\mathbf{I}}$
avg
$\{\psi_k^t\}$
{..., Rabbit, ...}
Con Mem
θ_{con}^t (init by $\theta_{\text{con}}^{t-1}$)
$f_{\mathbf{T}}$
f_{VL}
Image-to-Text Attn
Text-to-Image Attn
θ_{inc}^t (init by $\theta_{\text{inc}}^{t-1}$)
$\{(\psi_k^t, \theta_{\text{con}}^t, \theta_{\text{inc}}^t)\}$
f_{dec}
No obj
No obj
Rabbit
...
Rabbit
$\mathcal{L}_{\text{cls}}$
$\mathcal{L}_{\text{reg}}$
Step t+1
t+1-th train set
Enc
VL Enc
Dec
Memory
$\{(\psi_k^{t+1}, \theta_{\text{con}}^{t+1}, \theta_{\text{inc}}^{t+1})\}$
Memory Pool $\mathbb{B}$
Test
$f_{\mathbf{I}}$
$\mathbb{B}$
$\mathbb{C}_{\text{old}} \cup \mathbb{C}_{\text{new}} \cup \mathbb{C}_{\text{unseen}}$
Con Mem
$\theta_{\text{con}}^{\text{opt}}$
$\{(\psi^{\text{opt}}, \theta_{\text{con}}^{\text{opt}}, \theta_{\text{inc}}^{\text{opt}})\}$
$f_{\mathbf{I}}$
$f_{\mathbf{T}}$
f_{VL}
I2T Attn
T2I Attn
$\theta_{\text{inc}}^{\text{opt}}$
f_{dec}
Supervision
Forward
Data Flow
LoRA
Learned Params

Figure 2. Overview of our proposed MR-GDINO. MR-GDINO is based on a frozen pretrained open-world object detector with explicit visual-language interaction modules (*e.g.*, Grounding DINO [34]). During each step t of training, MR-GDINO initializes concept memory θ_{con}^t and visual-language interaction memory θ_{inc}^t from corresponding parameters in the $t-1$ step, and optimizes both parameters by t-th training set. After training, θ_{con}^t and θ_{inc}^t are memorized into the memory pool $\mathbb{B}$. During open-world inference scenarios, MR-GDINO uses the global embedding of input image $\mathbf{I}$ to retrieve the optimal parameters $(\psi^{\text{opt}}, \theta_{\text{con}}^{\text{opt}}, \theta_{\text{inc}}^{\text{opt}})$ and use these parameters for accurate predictions.

and seen images are usually fully labeled [32, 44, 52] during annotation. MR-GDINO avoids such irregular scenario.Besides, with the growing use of OW models in continual learning [55, 59], OW-COD emphasizes anti-forgetting capabilities for unseen categories.

Comparison with OWOD. OWOD [2, 29, 40, 50, 63] can be seen as a zero-shot special case of our task. In contrast, OW-COD simultaneously emphasizes the anti-forgetting capabilities among old, new, and unseen categories, which largely requires the generalization of OW detectors.

Comparison with Deng *et al.* Deng *et al.*[8] conducted initial studies on open-world continual learning. However, their approach has two main drawbacks. First, the task-incremental evaluation is impractical for real-world applications and oversimplifies the challenge for continual open-world (OW) detectors. Second, their use of COCO [32] for OW evaluation is limited, as it contains only 80 common classes that frequently recur across continual learning steps, thereby reducing the task's complexity.

4. Proposed Method

4.1. Overview of MR-GDINO

To accomplish the goal of OW-COD, our core idea is *first leveraging parameter-efficient modules to formulate "memories" in each step, then adaptively retrieving optimal memory for robust performance*. Therefore, we propose MR-GDINO, a strong baseline built upon the OW-COD benchmark. MR-GDINO utilizes a frozen open-world object detector with explicit visual-language interaction modules [29, 34] (*e.g.*, Grounding DINO [34]) and incorporates memory and retrieval mechanisms for detection. The training and testing pipeline of MR-GDINO is illustrated in Fig. 2. Specifically, during training of step t, given input image $\mathbf{I}$ and corresponding training label set $\mathbb{C}_t$, MR-GDINO first concatenates class names in $\mathbb{C}_t$ by dot symbol, and formulate a unified class sentence $\mathbf{T}_t$. Then MR-GDINO calculates dense image feature $\mathbf{F_I}$ and text feature $\mathbf{F_T}$ by image feature extractor $f_{\mathbf{I}}(\cdot;\theta_{\mathbf{I}})$ and text feature extractor $f_{\mathbf{T}}(\cdot;\theta_{\mathbf{T}},\theta_{\text{con}}^t)$ respectively, , where θ_{con}^t is the parameters of our proposed concept memory. Next $\mathbf{F_I}$ and $\mathbf{F_T}$ are fed into the visual-language feature enhancer $f_{\mathbf{VL}}(\cdot;\theta_{\mathbf{VL}},\theta_{\text{inc}}^t)$ and obtain refined features $\mathbf{F}_{\mathbf{I}}^{'}$ and $\mathbf{F}_{\mathbf{T}}^{'}$, where θ_{inc}^t is the parameter of our proposed VL interaction memory. Finally, $\mathbf{F}_{\mathbf{I}}^{'}$ and $\mathbf{F}_{\mathbf{T}}^{'}$ are fed into the visual-language decoder $f_{\text{dec}}(\cdot;\theta_{\text{dec}})$ and obtain per-object detection results. Such results are supervised by corresponding ground-truth and used to optimize θ_{con}^t and θ_{inc}^t. And during inference, the input image $\mathbf{I}$ first extract the global embedding $\mathbf{g_I}$ by image feature extractor $f_{\mathbf{I}}$. Then MR-GDINO uses $\mathbf{g_I}$ as query to retrieve the optimal memory triplets $\{(\psi^{\text{opt}},\theta_{\text{con}}^{\text{opt}},\theta_{\text{inc}}^{\text{opt}})\}$ from the memory pool $\mathbb{B}$ by threshold τ. Finally, both input image $\mathbf{I}$ and class sentence $\mathbf{T}$ are fed into each OW detector $f(\cdot,\theta_f,\theta_{\text{inc}}^{\text{opt}})$ for initial detection results. These results are post-processed by

Non-Maximum Suppression (NMS) [19] for final results.

4.2. Concept and Interaction Memory Mechanism

Inspired by parameter-efficient fine-tuning techniques in few-shot learning [9, 24, 66] and continual learning [10, 55, 59], MR-GDINO utilizes parameter-efficient modules as memory units (*i.e.*, concept memory and visual-language (VL) interaction memory) for continually added classes to build optimal memories in corresponding learning steps.

Concept memory. To make $f_{\mathbf{T}}(\cdot;\theta_{\mathbf{T}})$ adapt to continually added classes with negligible extra parameters, we introduce a learnable prompt θ_{con} into $f_{\mathbf{T}}$. During the t-th training step, with given class sentence $\mathbf{T}$, MR-GDINO first convert $\mathbf{T}$ to initial text embedding $\mathbf{E}$ via embedding layer, and then concatenates both $\mathbf{E}$ and θ^t_{con}, finally the concatenated sequences are fed into transformer blocks in $f_{\mathbf{T}}$ and obtain the final text embedding $\mathbf{F_T}$.

VL interaction memory. Inspired by explicit VL interaction modules [29, 34], we conclude that enhancing VL interaction of each step on these modules can lead to better continual OW detectors. To retrieve the optimal memory from memory pool for mitigating catastrophic forgetting, we propose VL interaction memory and leverage LoRA [21] as corresponding memory, as shown in Fig. 3. In each j-th layer of $f_{\mathbf{VL}}$, given $\mathbf{F_I}$ and $\mathbf{F_T}$, MR-GDINO uses deformable self-attention [67] and vanilla self-attention [48] to refine image and text features respectively, thereby obtaining $\hat{\mathbf{F}}_{\mathbf{I}}$ and $\hat{\mathbf{F}}_{\mathbf{T}}$. Then MR-GDINO calculates aggregated text feature $\tilde{\mathbf{F}}_{\mathbf{T}}$ by:

$$\begin{aligned}
\tilde{\mathbf{F}}_{\mathbf{T}} &= \text{Attn}(\mathbf{q_T}, \mathbf{k_I}, \mathbf{v_I}), \text{ where} \\
\mathbf{q_T} &= (\mathbf{Q_{I\to T}} + \mathbf{B^q_{I\to T}}\mathbf{A^q_{I\to T}})\hat{\mathbf{F}}_{\mathbf{T}} \\
\mathbf{k_I} &= (\mathbf{K_{I\to T}} + \mathbf{B^k_{I\to T}}\mathbf{A^k_{I\to T}})\hat{\mathbf{F}}_{\mathbf{I}} \\
\mathbf{v_I} &= (\mathbf{V_{I\to T}} + \mathbf{B^v_{I\to T}}\mathbf{A^v_{I\to T}})\hat{\mathbf{F}}_{\mathbf{I}}
\end{aligned} \tag{3}$$

where "Attn" means cross-attention [48], $\mathbf{A}$ and $\mathbf{B}$ represent LoRA down- and up-projection layers. Note that $\mathbf{Q}$ and $\mathbf{K}$ with corresponding LoRA share the same parameters, and only $\mathbf{A}$ and $\mathbf{B}$ are optimized during training. Next, MR-GDINO calculates the aggregated image feature $\tilde{\mathbf{F}}_{\mathbf{I}}$ by:

$$\begin{aligned}
\tilde{\mathbf{F}}_{\mathbf{I}} &= \text{Attn}(\mathbf{q_I}, \mathbf{k_T}, \mathbf{v_T}), \text{ where} \\
\mathbf{q_I} &= (\mathbf{Q_{T\to I}} + \mathbf{B^q_{T\to I}}\mathbf{A^q_{T\to I}})\hat{\mathbf{F}}_{\mathbf{I}} \\
\mathbf{k_T} &= (\mathbf{K_{T\to I}} + \mathbf{B^k_{T\to I}}\mathbf{A^k_{T\to I}})\tilde{\mathbf{F}}_{\mathbf{T}} \\
\mathbf{v_T} &= (\mathbf{V_{T\to I}} + \mathbf{B^v_{T\to I}}\mathbf{A^v_{T\to I}})\tilde{\mathbf{F}}_{\mathbf{T}}
\end{aligned} \tag{4}$$

Finally, $\tilde{\mathbf{F}}_{\mathbf{I}}$ and $\tilde{\mathbf{F}}_{\mathbf{T}}$ are refined by corresponding feed-forward networks. After L-layer aggregation, one can obtain the final $\mathbf{F}'_{\mathbf{I}}$ and $\mathbf{F}'_{\mathbf{T}}$ for object detection. And the learned $\mathbf{A}$ and $\mathbf{B}$ in all layers formulate θ^t_{inc} in the t-th step.

4.3. Memory Retrieval Mechanism

Both kind of memories can effectively incorporate knowledge regarding each step. Nevertheless, such memories still faces catastrophic forgetting in both unseen and specific previously learned scenarios. To mitigate this issue, an intuitive idea is explicitly memorizing all previously learned memories, and adaptively retrieving best matched modules during inference. Such method is also aligned with human memory [1, 6]. Therefore, we propose the retrieval mechanism. Specifically, MR-GDINO introduces a memory pool $\mathbb{B}$ to store all previously learned memories. For the t-th step, with given n-shot training images, MR-GDINO first augment images by instance cropping and obtain totally N images for each class k in $\mathbb{C}_t$. Then MR-GDINO utilizes $f(\cdot;\theta_I)$ to calculate each averaged global embedding ψ^t_k averaging operation. And finally, MR-GDINO formulates the triplet of t-th step by $\{(\psi^t_k, \theta^t_{\text{con}}, \theta^t_{\text{inc}})\}$, where θ^t_{con} and θ^t_{inc} are optimized concept and VL interaction memory from t-th step. And during inference, given input image $\mathbf{I}$, MR-GDINO first extract the global embedding $\mathbf{g_I}$, then we find the indices $\{\hat{t}\}$ of memory triplets by threshold τ, and ensure that $\{\hat{t}\} = \{t|\langle \mathbf{g_I}, \psi^t\rangle > \tau\}$. Finally, we retrieve the optimal memories $\{(\psi^{\text{opt}}, \theta^{\text{opt}}_{\text{con}}, \theta^{\text{opt}}_{\text{inc}})\}$ by:

$$(\psi^{\text{opt}}, \theta^{\text{opt}}_{\text{con}}, \theta^{\text{opt}}_{\text{inc}}) = \begin{cases} \{(\psi^{\hat{t}}, \theta^{\hat{t}}_{\text{con}}, \theta^{\hat{t}}_{\text{inc}})\} & \langle \mathbf{g_I}, \psi^{\hat{t}}\rangle \geqslant \tau \\ \phi & \{\hat{t}\} = \phi \end{cases} \tag{5}$$

Such design ensures that, when unseen objects occurs, MR-GDINO enables to use the vanilla pretrained OW detector to detect objects in the wild, thus preserving the detection abilities of open-world unseen categories.

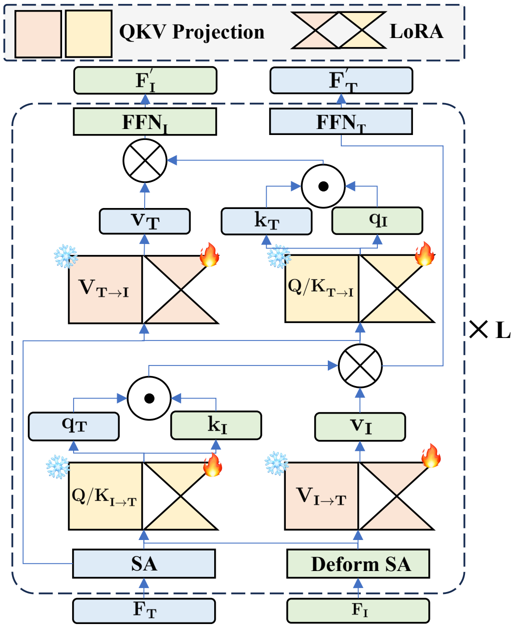


Figure 3. Overview of the proposed visual-language interaction memory. Specifically, MR-GDINO adopts LoRA [21] modules as θ_{inc} in Q/K/V projections of VL feature enhancer $f_{\mathbf{VL}}$.

4.4. Training of MR-GDINO

During training, parameters of pretrained OW detector are frozen to preserve robust feature representation [9, 24, 55],

Table 1. Comparison between MR-GDINO and counterparts. MR-GDINO simultaneously merits from flexibility, scalability, efficiency, thus achieving better anti-forgetting ability.

Method	Flexibility	Scalability	Efficiency	OW Anti-forget
CoOp [66]	✗	✗	✓	✗
L2P [55]	✓	✗	✓	✗
ZiRa [8]	✓	✗	✓	✗
CL-DETR [35]	✗	✗	✗	✗
MR-GDINO	✓	✓	✓	✓

while only concept and VL interaction memories are optimized. Specifically, to maintain a consistent text embedding distribution from frozen $f_{\mathbf{T}}$ for stable training, memory training is divided into two stages. In the first stage, MR-GDINO freezes VL interaction memory and optimizes concept memory to adapt to new classes. In the second stage, the updated concept memory is frozen, and interaction memory is optimized to refine visual-language relationships. Notably, joint training of both memory types can achieve similar performance, as discussed in Sec. 5.4.

Training Objectives. Unlike previous works [12, 35], MR-GDINO does not use additional losses specifically designed for continual learning. For bounding box regression, MR-GDINO minimizes L1 loss and GIoU loss [43] at each training step. For object classification, focal loss [33] is employed to enhance recognition performance.

4.5. Relation with Counterparts and Merits

As shown in Table 1, MR-GDINO excels in three aspects. For **flexibility**, it outperforms CoOp and CL-DETR with flexible memory retrieval through activated parameter selection. For **scalability**, MR-GDINO surpasses L2P [55] with a scalable memory pool that preserves and integrates knowledge. Lastly, for **efficiency**, MR-GDINO leverages parameter-efficient fine-tuning, outperforming traditional full fine-tuning methods [4, 35, 64]. These strengths ensure strong performance on old, new, and unseen classes.

5. Experiments

We compare MR-GDINO with zero-shot GDINO [34], CoOp [66], L2P [55], Adapter [20], and ZiRa [8]. All methods are designed for continual or fast adaptation.

5.1. Implementation Details

We employ the Swin-T [36] Grounding DINO [34] as the pretrained OW detector for both MR-GDINO and the counterparts. For continual training on OW-COD, we optimize OW detectors following the ascending dictionary order of subsets and evaluate the trained detectors on old, new, and unseen categories from corresponding subsets without any test-time augmentation. For MR-GDINO, we set a default prompt length of 10 and a LoRA [21] bottleneck dimension of 8. We use AdamW [38] with cosine learning rate scheduler [15, 37] to optimize MR-GDINO with weight decay of 1e-2 and batch size of 1 per GPU. The initial learning rate candidates are {1e-1, 4e-2, 1e-2, 1e-3, 1e-4}, and training epochs range from {1∼10}. We perform grid search [27] to find optimal hyper-parameters for each step. τ is set to 0.89 by default. Baseline methods are constructed and optimized using their default hyper-parameters. Due to the absence of an LVIS evaluation toolkit in the original GDINO implementation, we implement corresponding toolkit to fairly assess old, new, and unseen classes across all methods.

5.2. Comparison with State-of-The-Arts

Table 2 presents the comparison between MR-GDINO and all the counterparts under continual adaptations with different shots. Among all the counterparts, only ZiRa [8] surpasses ZS GDINO by 3.1 on AP^{seen} after 10-shot continual adaptations, while other methods fail to outperform GDINO. For unseen classes, only the Adapter [20] based continual OW detector achieves comparable albeit lower mAP, with all other methods suffering significant catastrophic forgetting. These findings strongly support our perspective and highlight the importance of OW-COD. In contrast, MR-GDINO under 10-shot training achieves 51.9 AP^{seen} and 20.7 $\text{AP}^{\text{unseen}}$. Moreover, even in 1-shot continual learning settings, MR-GDINO still achieves 46.7 seen mAP and only suffers 0.1 drop in terms of unseen mAP, and still largely surpasses all the counterparts on both metrics. Such promising results demonstrate that MR-GDINO can largely improves the detection performance on old and new classes, meanwhile maintaining robust detection abilities for unseen categories. We also investigate corresponding forgetting rate in each training step, which is listed in the suppl. Furthermore, though ZiRa and Adapter show improved anti-forgetting abilities for seen and unseen categories, respectively, their average rankings remain affected by the imbalanced performance between seen and unseen classes. In contrast, MR-GDINO achieves rank 1.3 in terms of R^{avg} on the leaderboard, underscoring its balanced and superior performance across old, new, and unseen classes.

Qualitative Results. Besides, we present qualitative results among ZS GDINO [34], ZiRa [8], and MR-GDINO, as shown in Fig. 4. Notably, MR-GDINO produces accurate bounding boxes with higher confidence for both old and new classes. Moreover, MR-GDINO outperforms ZiRa in generating accurate bounding boxes for unseen classes. These results further confirm the effectiveness of MR-GDINO. More qualitative results are shown in the suppl.

5.3. MR-GDINO Can Mitigate Forgotten Classes

Based on the promising anti-forgetting capabilities in both seen and unseen classes, one can leverage MR-GDINO to mitigate "forgotten" classes from fine-tuning. Specifically, we fully fine-tune GDINO [34] on COCO [32], and corre-

Table 2. Comparison of diverse open-world continual learning frameworks . We keep the pretrained models of all frameworks are the same Grounding DINO with Swin-T. Best results are **bolded** and second best results are underlined. Compared to zero-shot GDINO, all the baselines face severe catastrophic forgetting on either seen classes or open-world unseen classes. In contrast, MR-GDINO expresses promising anti-forgetting capabilities on both seen and unseen classes, and surpasses all the counterparts in terms of detection abilities.

Shot	**Method**	**Ae**	**Aq**	**Co**	**Eg**	**Mu**	**Pa**	**VOC**	**Pi**	**Po**	**Ra**	**Sh**	**Th**	**Ve**	**Seen**	**Unseen**	R^{avg}
0	ZS GDINO	15.4	18.4	69.0	57.7	14.8	65.3	50.2	53.8	16.1	0	22.8	42.6	58.8	37.3	20.7	-
1	CoOp	14.6	15.7	73.6	0	58.1	63.3	**54.2**	**49.7**	9.6	27.6	**31.9**	57.5	**59.6**	39.6	20.5	2.1
	Adapter	13.7	17.6	66.2	55.8	44.9	67.4	48.4	41.4	12.3	0	20.8	45.7	55.5	37.7	19.7	3.3
	L2P	11.5	16.9	71.0	2.5	45.2	53.6	45.3	48.4	3.0	25.2	22.2	37.8	59.4	34.0	18.7	3.9
	ZiRa	11.4	12.4	0	9.8	**66.1**	68.6	39.1	32.7	2.0	**48.7**	23.4	57.1	50.4	32.5	6.9	4.4
	MR-GDINO	**20.0**	**20.8**	**76.2**	**59.6**	56.3	**69.0**	51.5	49.3	**19.7**	26.3	28.4	**71.1**	58.9	**46.7**	**20.6**	**1.3**
3	CoOp	10.5	3.7	0	0	30.6	59.9	8.3	39.5	1.4	12.8	14.3	32.1	47.1	20.1	19.4	3.6
	Adapter	14.8	18.4	68.9	55.7	47.4	67.4	49.2	39.9	12.5	0	22.6	51.4	58.0	39.0	19.4	2.4
	L2P	13.2	15.8	74.3	15.3	36.5	66.4	50.0	43.8	2.1	8.5	18.7	46.6	**63.1**	35.0	18.8	3.6
	ZiRa	13.4	14.3	0	2.6	50.0	60.7	47.1	51.2	7.3	**52.9**	**33.4**	56.6	42.4	33.2	7.3	4.2
	MR-GDINO	**28.6**	**26.2**	**76.5**	**67.9**	**73.1**	**68.2**	**50.9**	**58.8**	**22.2**	30.8	25.6	**70.6**	59.5	**50.7**	**20.6**	**1.1**
5	CoOp	9.8	12.5	42.2	0	56.2	55.1	28.2	22.6	3.9	30.0	25.7	28.0	61.3	28.9	19.1	3.5
	Adapter	14.4	18.7	69.3	56.8	47.1	67.3	49.7	41.7	13.1	0	23.1	49.0	57.6	39.1	20.2	2.5
	L2P	11.3	14.3	50.3	0	42.8	59.9	35.8	52.1	3.9	27.2	23.8	35.3	**64.4**	32.4	17.4	3.7
	ZiRa	12.5	9.2	44.5	0	38.0	59.7	47.8	55.9	4.4	34.9	**30.2**	**56.6**	62.9	35.1	5.8	4.1
	MR-GDINO	**28.7**	**26.3**	**80.6**	**69.4**	**60.1**	**74.0**	**50.8**	**63.8**	**25.4**	**43.3**	23.6	52.7	62.0	**50.8**	**20.6**	**1.3**
10	CoOp	11.9	16.3	56.1	0.4	57.6	59.5	44.0	45.3	4.9	19.1	23.6	46.6	62.7	34.5	17.0	3.7
	Adapter	16.8	18.1	71.8	54.7	37.0	66.1	**50.5**	35.4	11.7	0	25.1	40.0	58.2	37.3	20.4	2.7
	L2P	9.1	12.9	22.8	0.9	41.3	50.3	30.3	41.0	11.7	9.0	19.0	37.8	61.8	26.8	17.8	3.9
	ZiRa	10.7	6.5	74.9	0	66.0	**69.8**	46.0	49.7	6.0	**40.3**	**32.4**	59.6	**63.2**	40.4	6.9	4.0
	MR-GDINO	**30.5**	**26.1**	**79.4**	**65.3**	**75.3**	67.1	48.3	**65.0**	**30.4**	27.4	25.7	**74.8**	59.7	**51.9**	**20.7**	**1.3**

Table 3. Comparison between Grounding DINO after COCO fully fine-tuning and that with MR-GDINO (10-shot), where red means subsets with performance drop after fine-tuning. Compared to zero-shot Grounding DINO and GDINO (COCO-ft), MR-GDINO enables to mitigate forgotten classes by few-shot continual adaptations, meanwhile preserving promising detection abilities on unseen classes.

Method	COCO	**Ae**	**Aq**	**Co**	**Eg**	**Mu**	**Pa**	**VOC**	**Pi**	**Po**	**Ra**	**Sh**	**Th**	**Ve**	**Seen**	**Unseen**
ZS GDINO	48.4	15.4	18.4	69.0	57.7	14.8	65.3	50.2	53.8	16.1	0.0	22.8	42.6	58.8	37.3	20.7
GDINO (COCO-ft)	**57.3**	12.5	20.7	64.1	34.8	43.6	**66.4**	52.0	48.5	7.2	0.0	35.0	46.0	54.2	37.3	23.6
+MR-GDINO	57.2	**20.7**	**29.2**	**82.1**	**64.8**	**83.3**	66.2	**62.2**	**65.1**	**16.2**	**50.2**	**40.2**	**69.8**	**59.6**	**54.5**	**23.6**

sponding evaluation results are shown in Table 3. Though detection performance on COCO increases to 57.3 mAP, detection performance on 6 out of 13 subsets has dropped, which can be seen as forgotten unseen classes. By adopting MR-GDINO onto GDINO (COCO-ft), detection performance on above subsets has increased and achieves 54.5 $\mathrm{AP}^{\mathrm{seen}}$. Meanwhile, since COCO [32] and LVIS [17] has large overlap in image domain, the $\mathrm{AP}^{\mathrm{unseen}}$ of GDINO (COCO-ft) has increased to 23.6 due to fully fine-tuning. Compared to GDINO (COCO-ft), that with MR-GDINO preserves the same $\mathrm{AP}^{\mathrm{unseen}}$. Above results further verify the effectiveness of MR-GDINO in mitigating forgetting.

5.4. Empirical Analysis

5.4.1. Ablation Study of Each Component

We first conduct ablation study of each component with 10-shot continual learning. Table 4 demonstrates the evaluation results of each method. After adopting θ_{con}, $\mathrm{AP}^{\mathrm{old}}$ and θ_{con}, $\mathrm{AP}^{\mathrm{unseen}}$ largely drops to 32.2 and 17.0 respectively, but $\mathrm{AP}^{\mathrm{new}}$ largely increases to 62.1. Similarly, when further adopting θ_{inc} into MR-GDINO, corresponding $\mathrm{AP}^{\mathrm{new}}$ increases to 63.1. Above optimized memories provide strong and robust learned parameters on each subset and will benefit retrieval mechanism. After adopting retrieval mechanism, both $\mathrm{AP}^{\mathrm{old}}$ and $\mathrm{AP}^{\mathrm{unseen}}$ significantly increase to 51.3 and 20.7 respectively, which indicates that such mechanism can effectively retrieve optimal θ_{con} and θ_{inc} for given inputs to achieve better detection abilities. And if the input images come from unseen categories, MR-GDINO can still execute correct action and use ZS GDINO for inference. These findings verify the effectiveness of memory and retrieval mechanisms in OW-COD, and reveal potential directions towards better continual OW detectors.

5.4.2. Effect of Inserted Layer Number for θ_{inc}

Next we investigate the effect from inserted layer number of θ_{inc}, corresponding results are shown in Table 5. By insert-

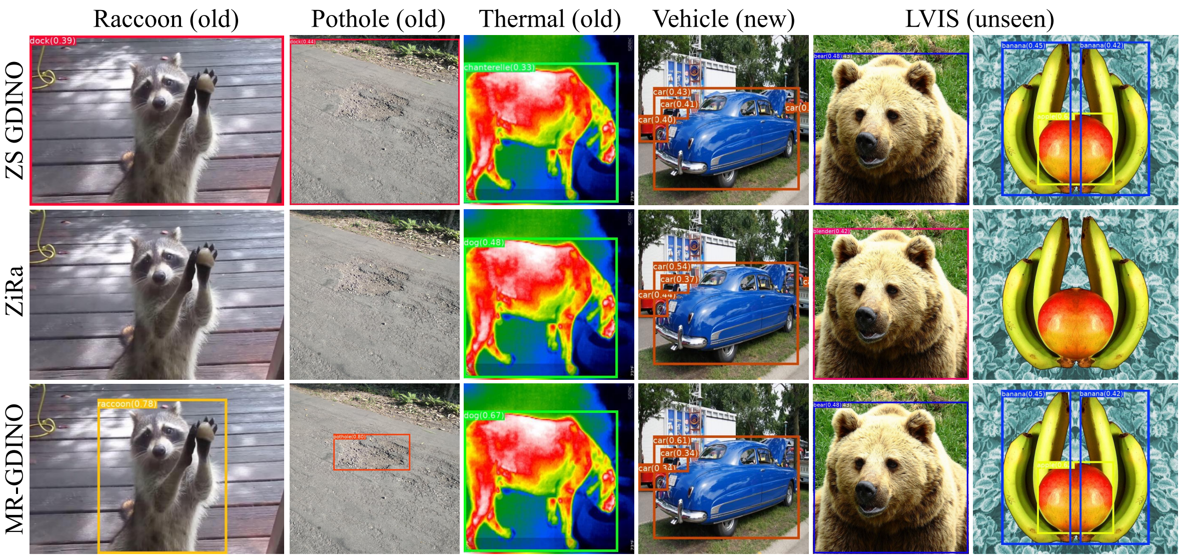


Figure 4. Qualitative results of zero-shot Grounding DINO [34], ZiRa [8], and MR-GDINO. Compared to ZS GDINO and state-of-the-art ZiRa, MR-GDINO can generate more accurate bounding boxes with higher confidence on both seen and unseen classes.

Table 4. Ablation study of key components in MR-GDINO.

GDINO	θ_{con}	θ_{inc}	retrieval	AP^{old}	AP^{new}	AP^{seen}	$\text{AP}^{\text{unseen}}$
✓	✗	✗	✗	35.5	58.8	37.3	20.7
✓	✓	✗	✗	32.2	62.1	34.5	17.0
✓	✓	✗	✓	44.4	56.8	45.4	20.7
✓	✓	✓	✗	30.9	63.1	33.3	10.7
✓	✓	✓	✓	51.3	59.7	51.9	20.7

Table 5. Number of θ_{inc}-inserted layers analysis.

Method	Layers	Added Params	AP^{old}	AP^{new}	AP^{seen}	$\text{AP}^{\text{unseen}}$
MR-GDINO	0	12K	44.4	56.8	45.4	20.7
MR-GDINO	1	53K	47.3	57.5	48.1	20.7
MR-GDINO	2	94K	49.0	58.2	49.7	20.7
MR-GDINO	3	135K	50.1	59.9	50.8	20.7
MR-GDINO	6 (all)	258K	51.3	59.7	51.9	20.7

Table 6. Comparison between joint and decoupled training.

Joint	Decouple	AP^{old}	AP^{new}	AP^{seen}	$\text{AP}^{\text{unseen}}$
✓	-	51.6	59.7	52.2	20.7
-	✓	51.3	59.7	51.9	20.7

Table 7. Performance gap between MR-GDINO and MR-GDINO with oracle retrieval module.

Retrieval	AP^{old}	AP^{new}	AP^{seen}	$\text{AP}^{\text{unseen}}$
MR-GDINO	51.3	59.7	51.9	20.7
Oracle	51.8	59.8	52.4	20.7

ing θ_{inc} in more layers, AP^{old} are gradually increased from 44.4 to 51.3, while maintaining the same $\text{AP}^{\text{unseen}}$. These results show that inserting θ_{inc} to more VL interaction layers lead to better performance with negligible parameters.

5.4.3. Decoupled Training or Joint Training

We also investigate whether MR-GDINO supports joint training for θ^t_{con} and θ^t_{inc} at each training step t. Using the optimal training hyper-parameters identified from decoupled training, we simultaneously optimize θ^t_{con} and θ^t_{inc}. The results, shown in Table 6, indicate that joint training achieves the same 59.7 AP^{new} and $\text{AP}^{\text{unseen}}$, with slightly improved AP^{old}. These findings suggest that once optimal hyper-parameters are confirmed, joint optimization can halve the training time to improve efficiency.

5.4.4. Performance Gap with Oracle Retrieval

Finally, we analyze the retrieval mechanism to assess the performance gap between MR-GDINO and oracle counterparts. For the oracle retrieval, we assign $\theta^{\text{opt}}_{\text{con}}$ and $\theta^{\text{opt}}_{\text{inc}}$ using ground-truth labels and report detection results in Table 7. Compared to the oracle, MR-GDINO shows a minor decrease of 0.5 in AP^{old} and 0.1 in AP^{new}, while achieving similar performance in $\text{AP}^{\text{unseen}}$. These results confirm the effectiveness of MR-GDINO's retrieval mechanism. However, exploring more precise retrieval mechanisms remains valuable for future large-scale and practical applications. Further analysis is provided in the supplementary material.

6. Conclusion

We propose open-world continual object detection, requiring detectors to generalize across old, new, and unseen categories. To evaluate OW detectors with existing continual learning methods, we propose OW-COD, a benchmark encouraging OW detectors to preserve old classes, adapt to new ones, and maintain open-world detection abilities. To address catastrophic forgetting of unseen categories, we propose a strong baseline namely MR-GDINO, a scalable open-world object detection framework utilizing memory and retrieval in a compact memory pool. Our results show that MR-GDINO minimizes catastrophic forgetting with only 0.1% additional parameters, achieving state-of-the-art performance on OW-COD.

References

[1] Alan D Baddeley. *Human memory: Theory and practice*. psychology press, 1997. 5

[2] Ankan Bansal, Karan Sikka, Gaurav Sharma, Rama Chellappa, and Ajay Divakaran. Zero-shot object detection. In *Proceedings of the European conference on computer vision (ECCV)*, pages 384–400, 2018. 1, 2, 3, 4

[3] Soravit Changpinyo, Piyush Sharma, Nan Ding, and Radu Soricut. Conceptual 12M: Pushing web-scale image-text pre-training to recognize long-tail visual concepts. In *CVPR*, 2021. 1, 2

[4] L. Chen, Chunyan Yu, and Lvcai Chen. A new knowledge distillation for incremental object detection. *2019 International Joint Conference on Neural Networks (IJCNN)*, pages 1–7, 2019. 6

[5] Tianrun Chen, Lanyun Zhu, Chaotao Ding, Runlong Cao, Yan Wang, Zejian Li, Lingyun Sun, Papa Mao, and Ying Zang. Sam fails to segment anything?–sam-adapter: Adapting sam in underperformed scenes: Camouflage, shadow, medical image segmentation, and more. *arXiv preprint arXiv:2304.09148*, 2023. 1

[6] Nelson Cowan. *Working memory capacity*. Psychology press, 2012. 5

[7] Jieren Deng, Haojian Zhang, Jianhua Hu, Xingxuan Zhang, and Yunkuan Wang. Class incremental robotic pick-and-place via incremental few-shot object detection. *IEEE Robotics and Automation Letters*, 8(9):5974–5981, 2023. 1

[8] Jieren Deng, Haojian Zhang, Kun Ding, Jianhua Hu, Xingxuan Zhang, and Yunkuan Wang. Zero-shot generalizable incremental learning for vision-language object detection. *NeurIPS*, 2024. 1, 2, 3, 4, 6, 8

[9] Bowen Dong, Pan Zhou, Shuicheng Yan, and Wangmeng Zuo. Lpt: Long-tailed prompt tuning for image classification. In *The Eleventh International Conference on Learning Representations*. 2, 3, 5

[10] Bowen Dong, Guanglei Yang, Wangmeng Zuo, and Lei Zhang. Consept: Continual semantic segmentation via adapter-based vision transformer. *arXiv preprint arXiv:2402.16674*, 2024. 5

[11] Bowen Dong, Pan Zhou, and Wangmeng Zuo. Lpt++: Efficient training on mixture of long-tailed experts. *arXiv preprint arXiv:2409.11323*, 2024. 3

[12] 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*, pages 543–551, 2023. 3, 6

[13] Zi-Yi Dou, Aishwarya Kamath, Zhe Gan, Pengchuan Zhang, Jianfeng Wang, Linjie Li, Zicheng Liu, Ce Liu, Yann LeCun, Nanyun Peng, et al. Coarse-to-fine vision-language pre-training with fusion in the backbone. *Advances in neural information processing systems*, 35:32942–32956, 2022. 2

[14] Tao Feng, Mang Wang, and Hangjie Yuan. Overcoming catastrophic forgetting in incremental object detection via elastic response distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9427–9436, 2022. 3

[15] P Goyal. Accurate, large minibatch sg d: training imagenet in 1 hour. *arXiv preprint arXiv:1706.02677*, 2017. 6

[16] Yiduo Guo, Bing Liu, and Dongyan Zhao. Online continual learning through mutual information maximization. In *Proceedings of the 39th International Conference on Machine Learning*, pages 8109–8126. PMLR, 2022. 2

[17] Agrim Gupta, Piotr Dollar, and Ross Girshick. Lvis: A dataset for large vocabulary instance segmentation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 5356–5364, 2019. 1, 2, 3, 7

[18] Junxian He, Chunting Zhou, Xuezhe Ma, Taylor Berg-Kirkpatrick, and Graham Neubig. Towards a unified view of parameter-efficient transfer learning. In *International Conference on Learning Representations*, 2022. 1, 2

[19] Jan Hosang, Rodrigo Benenson, and Bernt Schiele. Learning non-maximum suppression. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 4507–4515, 2017. 5

[20] 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 *ICML*, 2019. 6

[21] Edward J Hu, yelong shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*, 2022. 1, 5, 6

[22] Zitong Huang, Ze Chen, Zhixing Chen, Erjin Zhou, Xinxing Xu, Rick Siow Mong Goh, Yong Liu, Wangmeng Zuo, and Chunmei Feng. Learning prompt with distribution-based feature replay for few-shot class-incremental learning. *arXiv preprint arXiv:2401.01598*, 2024. 3

[23] Zitong Huang, Ze Chen, Yuanze Li, Bowen Dong, Erjin Zhou, Yong Liu, Rick Siow Mong Goh, Chun-Mei Feng, and Wangmeng Zuo. Class balance matters to active class-incremental learning. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 9445–9454, 2024. 2

[24] Menglin Jia, Luming Tang, Bor-Chun Chen, Claire Cardie, Serge Belongie, Bharath Hariharan, and Ser-Nam Lim. Visual prompt tuning. In *European Conference on Computer Vision (ECCV)*, 2022. 5

[25] 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*, pages 1780–1790, 2021. 1

[26] Sahar Kazemzadeh, Vicente Ordonez, Mark Matten, and Tamara Berg. Referitgame: Referring to objects in photographs of natural scenes. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 787–798, 2014. 1, 2

[27] Steven M LaValle, Michael S Branicky, and Stephen R Lindemann. On the relationship between classical grid search

and probabilistic roadmaps. *The International Journal of Robotics Research*, 23(7-8):673–692, 2004. 6

[28] Feng Li, Hao Zhang, Peize Sun, Xueyan Zou, Shilong Liu, Jianwei Yang, Chunyuan Li, Lei Zhang, and Jianfeng Gao. Semantic-sam: Segment and recognize anything at any granularity. *arXiv preprint arXiv:2307.04767*, 2023. 1

[29] Liunian Harold Li*, Pengchuan Zhang*, Haotian Zhang*, Jianwei Yang, Chunyuan Li, Yiwu Zhong, Lijuan Wang, Lu Yuan, Lei Zhang, Jenq-Neng Hwang, Kai-Wei Chang, and Jianfeng Gao. Grounded language-image pre-training. In *CVPR*, 2022. 1, 2, 3, 4, 5

[30] Yiming Li, Yi Wang, Wenqian Wang, Dan Lin, Bingbing Li, and Kim-Hui Yap. Open world object detection: A survey. *IEEE Transactions on Circuits and Systems for Video Technology*, 2024. 1, 2

[31] Zhizhong Li and Derek Hoiem. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12):2935–2947, 2017. 3

[32] 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*, pages 740–755. Springer, 2014. 1, 3, 4, 6, 7

[33] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollar. Focal loss for dense object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 42(2):318–327, 2020. 6

[34] Shilong Liu, Zhaoyang Zeng, Tianhe Ren, Feng Li, Hao Zhang, Jie Yang, Chunyuan Li, Jianwei Yang, Hang Su, Jun Zhu, et al. Grounding dino: Marrying dino with grounded pre-training for open-set object detection. *ECCV*, 2024. 1, 2, 3, 4, 5, 6, 8

[35] 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*, pages 23799–23808, 2023. 3, 6

[36] Ze Liu, Yutong Lin, Yue Cao, Han Hu, Yixuan Wei, Zheng Zhang, Stephen Lin, and Baining Guo. Swin transformer: Hierarchical vision transformer using shifted windows. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 10012–10022, 2021. 6

[37] Ilya Loshchilov and Frank Hutter. SGDR: Stochastic gradient descent with warm restarts. In *International Conference on Learning Representations*, 2017. 6

[38] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2019. 6

[39] Jun Ma, Yuting He, Feifei Li, Lin Han, Chenyu You, and Bo Wang. Segment anything in medical images. *Nature Communications*, 15:654, 2024. 1

[40] Zongyang Ma, Guan Luo, Jin Gao, Liang Li, Yuxin Chen, Shaoru Wang, Congxuan Zhang, and Weiming Hu. Open-vocabulary one-stage detection with hierarchical visual-language knowledge distillation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 14074–14083, 2022. 2, 3, 4

[41] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning*, pages 8748–8763. PMLR, 2021. 1, 2

[42] Sylvestre-Alvise Rebuffi, Alexander Kolesnikov, Georg Sperl, and Christoph H Lampert. icarl: Incremental classifier and representation learning. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*, pages 2001–2010, 2017. 3

[43] Hamid Rezatofighi, Nathan Tsoi, JunYoung Gwak, Amir Sadeghian, Ian Reid, and Silvio Savarese. Generalized intersection over union. 2019. 6

[44] 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 *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 8429–8438, 2019. 1, 2, 4

[45] Konstantin Shmelkov, Cordelia Schmid, and Karteek Alahari. Incremental learning of object detectors without catastrophic forgetting. In *Proceedings of the IEEE international conference on computer vision*, pages 3400–3409, 2017. 3

[46] Xiang Song, Yuhang He, Songlin Dong, and Yihong Gong. Non-exemplar domain incremental object detection via learning domain bias. In *Proceedings of the AAAI Conference on Artificial Intelligence*, pages 15056–15065, 2024. 3

[47] Lv Tang, Haoke Xiao, and Bo Li. Can sam segment anything? when sam meets camouflaged object detection. *arXiv preprint arXiv:2304.04709*, 2023. 1

[48] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008, 2017. 5

[49] Eli Verwimp, Kuo Yang, Sarah Parisot, Lanqing Hong, Steven McDonagh, Eduardo Pérez-Pellitero, Matthias De Lange, and Tinne Tuytelaars. Clad: A realistic continual learning benchmark for autonomous driving. *Neural Networks*, 161:659–669, 2023. 2, 3

[50] 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 arXiv:2407.07844*, 2024. 1, 2, 3, 4

[51] Jianren Wang, Xin Wang, Yue Shang-Guan, and Abhinav Gupta. Wanderlust: Online continual object detection in the real world. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 10829–10838, 2021. 2, 3

[52] Jiaqi Wang, Pan Zhang, Tao Chu, Yuhang Cao, Yujie Zhou, Tong Wu, Bin Wang, Conghui He, and Dahua Lin. V3det: Vast vocabulary visual detection dataset. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 19844–19854, 2023. 1, 2, 4

[53] Xin Wang, Thomas Huang, Joseph Gonzalez, Trevor Darrell, and Fisher Yu. Frustratingly simple few-shot object de-

tection. In *International Conference on Machine Learning*, pages 9919–9928. PMLR, 2020. 3

[54] Xudong Wang, Shufan Li, Konstantinos Kallidromitis, Yusuke Kato, Kazuki Kozuka, and Trevor Darrell. Hierarchical open-vocabulary universal image segmentation. In *Thirty-seventh Conference on Neural Information Processing Systems*, 2023. 1

[55] Zifeng Wang, Zizhao Zhang, Chen-Yu Lee, Han Zhang, Ruoxi Sun, Xiaoqi Ren, Guolong Su, Vincent Perot, Jennifer Dy, and Tomas Pfister. Learning to prompt for continual learning. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 139–149, 2022. 1, 3, 4, 5, 6

[56] Lewei Yao, Jianhua Han, Youpeng Wen, Xiaodan Liang, Dan Xu, Wei Zhang, Zhenguo Li, Chunjing Xu, and Hang Xu. Detclip: Dictionary-enriched visual-concept paralleled pre-training for open-world detection. *Advances in Neural Information Processing Systems*, 35:9125–9138, 2022. 2

[57] Lewei Yao, Runhui Huang, Lu Hou, Guansong Lu, Minzhe Niu, Hang Xu, Xiaodan Liang, Zhenguo Li, Xin Jiang, and Chunjing Xu. FILIP: Fine-grained interactive language-image pre-training. In *International Conference on Learning Representations*, 2022. 1

[58] Lewei Yao, Jianhua Han, Xiaodan Liang, Dan Xu, Wei Zhang, Zhenguo Li, and Hang Xu. Detclipv2: Scalable open-vocabulary object detection pre-training via word-region alignment. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23497–23506, 2023. 2

[59] Jiazuo Yu, Yunzhi Zhuge, Lu Zhang, Ping Hu, Dong Wang, Huchuan Lu, and You He. Boosting continual learning of vision-language models via mixture-of-experts adapters. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 23219–23230, 2024. 4, 5

[60] Licheng Yu, Patrick Poirson, Shan Yang, Alexander C Berg, and Tamara L Berg. Modeling context in referring expressions. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part II 14*, pages 69–85. Springer, 2016. 1, 2

[61] 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. 1, 3

[62] Xiaohua Zhai, Basil Mustafa, Alexander Kolesnikov, and Lucas Beyer. Sigmoid loss for language image pre-training. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 11975–11986, 2023. 1

[63] Haotian Zhang, Pengchuan Zhang, Xiaowei Hu, Yen-Chun Chen, Liunian Li, Xiyang Dai, Lijuan Wang, Lu Yuan, Jenq-Neng Hwang, and Jianfeng Gao. Glipv2: Unifying localization and vision-language understanding. *Advances in Neural Information Processing Systems*, 35:36067–36080, 2022. 1, 2, 3, 4

[64] Jichuan Zhang, Wei Li, Shuang Cheng, Ya-Li Li, and Shengjin Wang. Dynamic object queries for transformer-based incremental object detection. *arXiv preprint arXiv:2407.21687*, 2024. 3, 6

[65] 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*, 2024. 1

[66] Kaiyang Zhou, Jingkang Yang, Chen Change Loy, and Ziwei Liu. Learning to prompt for vision-language models. *International Journal of Computer Vision*, 2022. 5, 6

[67] Xizhou Zhu, Weijie Su, Lewei Lu, Bin Li, Xiaogang Wang, and Jifeng Dai. Deformable {detr}: Deformable transformers for end-to-end object detection. In *International Conference on Learning Representations*, 2021. 5