GCD: Advancing Vision-Language Models for Incremental Object Detection via Global Alignment and Correspondence Distillation -Supplementary-

More ablation results

Ablation study for pesudo-labels threshold. In Tab. 1, we provide ablation for pseudo labels threshold p. Pseudo labels selection is a trade-off, where a low threshold means high recall but more noise, a high threshold will introduce less noise but many low-confidence samples are ignored. The best performance is acquired at $th \geq 0.4$, and the performance will drop if the threshold is set higher. As the precision and recall is hard to balance, we simply choose the threshold with highest Average Precision(AP).

Ablation study for τ of KL-divergence. In Tab. 2, we provide ablation for temperature factor τ which is used in Correspondence Response Distillation(CRD)'s alignment component to soften teacher's probabilities. The best performance of CRD is acquired at $\tau = 100$. We observe that a small temperature will lead to performance drop, indicating that distilling a soft probabilities is beneficial.

Ablation study for hyper-parameters. In Tab. 3, we provide ablation results for weights of CRD and CTD. For CRD's weight γ, we observe that performance variation is within 0.1, demonstrating the robustness of CRD. A higher γ means less forgetting for old knowledge but will also influence the plasticity of new knowledge, thus we set $\gamma = 1$ by default. For CTD, we have tested several different combinations. "Object Topology" means that we only distill topological relationships of semantic queries' prototypes within each mini-batch, similarly for "Text Topology". Setting $\lambda_1 = 3$, we acquire the best performance 46.5% for object topology loss. The combination of both topology make the best performance 46.7%, where we set $\lambda_1 = 3, \lambda_2 = 5$.

Fine-grained ablation study for main components. In Tab. 4, we present fine-grained ablation results for each component of our method. In row 3, we remove the α factor in CRD, causing all predictions to be distilled equally. This leads to background responses dominating the distillation process, significantly reducing CRD's effectiveness. In row 5, we remove the α weight in CTD, making all semantic queries contribute equally to estimating class-wise prototypes. This equal treatment of queries can cause inconsistencies between the teacher's and student's prototypes, weakening the efficacy of CTD.

Row	Setting	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
1	$th{\geq}0.2$	42.3	59.2	45.7	26.8	45.5	55.3
2	$th{\geq}0.3$	44.4	60.1	48.1	26.3	48.0	60.6
3	$th{\geq}0.4$	**45.3**	**62.1**	**49.2**	**28.7**	**48.7**	**59.8**
4	$th{\geq}0.5$	43.4	59.7	47.0	27.0	46.3	58.0

Table 1: Ablation result $(AP\%)$ for different thresholds (th) for selecting pseudo labels on COCO 2017 using the $70+10$ setting.

Row	Setting	AP	AP_{50}	AP_{75}	AP_S	AP_M	AP_L
1	$\tau = 1$	45.6	62.5	49.4	28.6	48.9	60.6
2	$\tau = 10$	45.8	62.8	49.6	28.8	49.1	60.7
3	$\tau = 50$	45.9	63.0	49.9	28.8	49.4	60.8
4	$\tau = 100$	**46.0**	**62.9**	**49.9**	**28.9**	**49.4**	**61.0**
5	$\tau = 200$	45.9	62.8	49.8	28.9	49.1	60.7

Table 2: Ablation result $(AP\%)$ for different temperature factor τ on COCO 2017 using the $70 + 10$ setting.

Row	Method		AP	FPP
1	CRD weights (γ)	$\gamma = 1$	**46.0**	**2.9**
		$\gamma = 3$	**46.0**	**2.9**
		$\gamma = 5$	45.9	2.8
2	Different CTD combination	Object Topology	46.5	2.2
		Text Topology	46.3	2.6
		Text + Object	**46.7**	**1.9**
3	CTD weights (λ_1, λ_2)	$(1, 0)$	46.1	2.8
		$(3, 0)$	46.5	2.2
		$(5, 0)$	46.2	2.6
		$(3, 3)$	46.6	2.0
		$(3, 5)$	**46.7**	**1.9**
		$(3, 10)$	46.5	1.9

Table 3: Ablation result $(AP\%)$ for different Hyper-parameters on COCO 2017 using the $70 + 10$ setting. Rows 1: weights γ of CRD's alignment part. Row 2: Different Topology combination used in CTD. Row3: weights (λ_1, λ_2) of CTD's object topology loss and text topology loss, respectively.

Row	Global	Pseudo	CRD (w/o α)	CRD	CTD (w/o α)	CTD	All categories ↑		Old categories ↑		FPP ↓	
							AP	AP_{50}	AP	AP_{50}	AP	AP_{50}
1	✓						6.3	8.4	1.7	2.4	48.3	65.2
2	✓	✓					45.3	62.1	46.3	63.7	3.7	3.9
3	✓	✓	✓				45.5	62.5	46.9	64.4	3.1	3.2
4	✓	✓		✓			46.0	63.0	47.1	64.6	2.9	3.0
5	✓	✓		✓	✓		46.1	63.1	47.4	64.8	2.6	2.8
6	✓	✓		✓		✓	**46.7**	**63.9**	**48.1**	**65.9**	**1.9**	**1.7**

Table 4: Fine grained ablation using COCO benchmark under $70 + 10$ setting. CRD(w/o α) represents α factor is discarded in CRD, similarly for CTD(w/o α). The best performance is highlighted in bold, with the final row indicating our method.

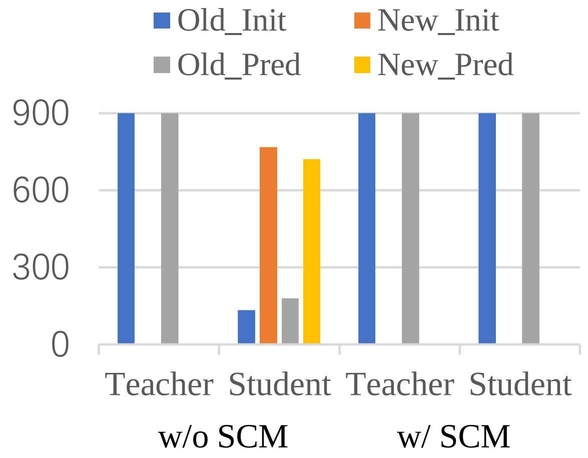


Figure 1: In the $70 + 10$ settings, we compare the teacher's and student's initialization and predictions. Y-axis represents the number of queries, total number is 900. "Old Init" refers to the classification results of the initialized queries corresponding to the categories from previous phase, while "Old Pred" refers to predictions belong to these categories. Similarly, "New Init" and "New Pred" refers to the initialization and predictions belong to new categories introduced in current phase. Additionally, "w/o SCM" indicates that the teacher and student initialize and make predictions separately, while "w/ SCM" refers to the introduction of shared queries and chunked text.

Ablation study for Semantic Correspondence Mechanism(SCM). In Fig. 1, we illustrate the query selection and final predictions for both the teacher and student models. The teacher uses $prompts_{1:t-1}$ to detect old objects, while the student uses $prompts_{1:t}$ to integrate new knowledge. Without SCM, the teacher selects queries based on the alignment score $S(V^{old}, W^{old})$, while the student selects queries based on $S(V^{global}, W^{global})$. We visualize the classification results of all initialized object queries and their final predictions. The student's query selection is heavily biased toward the new class, whereas the teacher is biased toward the old class. As a result, directly matching the teacher's and student's predictions is challenging. Introducing SCM helps mitigate this issue by ensuring distillation is performed on corresponding spatial and semantic relations.

Ablation study for Correspondence Response Distillation(CRD). In Fig. 2, we show the performance of weak categories, where "w/o CRD" represents using only global alignment, and "w/ CRD" indicates the combination of global alignment and CRD. The results show a clear improvement in weak categories with CRD. This is intuitive, as we set a high threshold to avoid noise, which causes weak categories to have prediction confidences below this threshold. As a result, these categories are not selected as pseudo-labels, leading to label conflicts and the collapse of their local semantic structure. CRD helps mitigate this by preserving the local semantic structure of these weak categories.

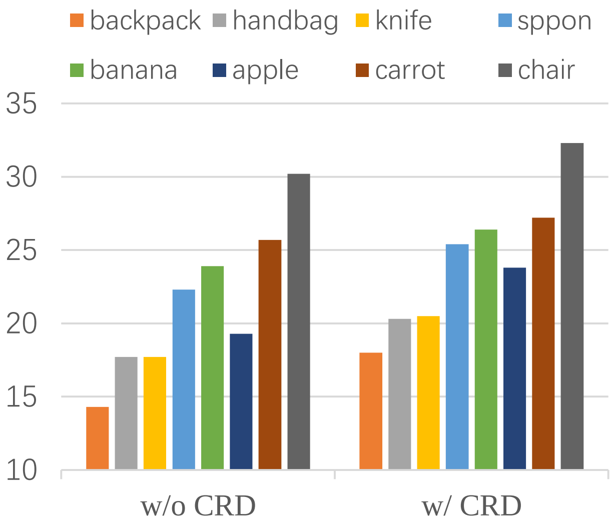


Figure 2: In $70 + 10$ setting, $AP(\%)$ of weak categories on w/o CRD and w/ CRD. With CRD, the AP of weak categories increase significantly. For example, backpack↑ 3.7%, handbag↑ 2.6%, knife ↑ 2.8%, spoon ↑ 3.1%, banana ↑ 2.5%, apple ↑ 4.5%, carrot ↑ 1.5%, chair ↑ 2.1%.

Ablation study for Correspondence Topology Distillation(CTD). In Fig. 3, we provide a T-SNE (Van der Maaten and Hinton 2008) visualization of the semantic query features for old categories under the $70 + 10$ setting. It is clear that local alignment leads to the collapse of the semantic structure of old knowledge due to the lack of constraints. We also show visualizations of (b) our method without CTD and (c) our complete method. In (b), without direct constraints on old knowledge at the feature level, integrating new knowledge causes the feature distribution to shift and rotate, which gradually disrupts the semantic structure of old knowledge. In (c), the introduction of CTD helps preserve the local semantic structure, mitigating these changes.

Method	Baseline model	AP(all)	AP(old)	AP(new)
Joint	G-DINO	49.6	50.6	42.2
Phase1	G-DINO	-	50.0	-
LWF	G-DINO	11.4	7.8	36.6
ERD	G-DINO	39.3	39.5	37.9
Ours	G-DINO	46.7	48.1	36.8
KD-oracle	G-DINO	47.2	48.2	40.5

Table 5: Comparison of KD-based methods ($AP\%$) on COCO 2017 in the 70+10 setting. ”Joint” represents fine-tuning on all 80 categories of COCO, while ”phase1” represents fine-tuning on the first 70 categories. LWF and ERD are reproduced based on G-DINO, and ”KD-oracle” indicates that complete annotations are provided in phase 2.

More comparison results of our methods

Comparison results of different KD-based methods. In Tab. 5, we provide comparison results of several KD-based methods. Here, Joint is the upper bound of G-DINO w/o O365 pretrain, where G-DINO is fine-tuned on COCO 2017 with complete training set. We also provide upper bound of phase 1, where G-DINO is fine-tuned on first 70 categories of COCO. Traditional response distillation methods LWF(Li and Hoiem 2017) and ERD(Feng, Wang, and Yuan 2022) are reproduced based on G-DINO. Additionally, we include the result of KD-oracle, where G-DINO is fine-tuned in phase 2 with full annotations. Typically, in phase 2 of $70{+}10$ setting, we only have access to annotations of last 10 categories. ”Full annotations” represent that we could access complete annotations within 80 categories. Compared to KD-oracle, our method is just 0.5% behind in AP (all categories) and only 0.1% behind in AP (old categories), demonstrating that our method effectively preserves old knowledge.

Computation costs of different methods. In Tab. 6, we present the computational costs of different methods during both training (on a single A100 GPU) and inference (on a single 3090 GPU). All methods are based on G-DINO-T and the batch is set 8 in training and 1 in inference. During training, our method offers a competitive balance between speed and memory usage by leveraging SCM to efficiently ensure the correspondence between the teacher's and student's predictions. In contrast, LWF and ERD require bipartite matching to establish correspondence, which is less memory efficient. At inference time, all methods are KD-based, introducing no additional computational overhead. Our approach seamlessly integrates new knowledge while preserving the capabilities of VLMs, maintaining nearly constant computational costs.

More Details of our methods

We provide a detailed description about our GCD method as illustrated in algorithm 1. All methods are integrated within the same pseudo code to directly reflect the actual execution process. We divide the forward process into three stages: encoding, query selection, and decoding. For example, receiving image x and $prompts_{1:t-1}$ for detecting old objects, the teacher model's encoding process transforms all input data into embeddings, specifically the image token V^{old} and text token W^{old}. In the query selection stage, V^{old} and W^{old} are used to generate a set of object queries, denoted as O^{old}. Finally, during the decoding process, the object queries O^{old}, along with image token V^{old} and text token W^{old}, are input to further refine coordinates and inject semantics. This results in high-level semantic queries $\hat{Q}_{old}$, which are then used to generate the final predictions $\hat{y}^{old}$. More details are provided in algorithm 1.

Method	Train(A100)		Inference(3090)		
	Speed	Memory	Speed	Memory	FLOPS
FT	5.33 FPS	26.7GB	6.25FPS	4.6GB	464G
LWF	1.78 FPS	41.2GB	6.25 FPS	4.6GB	464G
ERD	2.66 FPS	35.8GB	6.25 FPS	4.6GB	464G
Ours	2.70 FPS	31.9GB	6.25 FPS	4.6GB	464G

Table 6: Computation costs of different methods during training and inference. The table compares the training and inference speeds (FPS, defined as the number of images processed per second per GPU, where higher is better), memory usage (in GB, lower is better), and FLOPS (lower is better) for Fine-Tuning (FT), LWF, ERD, and our method on A100 and 3090 GPUs.

More visualization results

To intuitively understand the effect of the Semantic Correspondence Mechanism (SCM), we provide a visualization of the final predictions from both the teacher and student models as shown in Fig. 4. For clarity, we only display a subset of bounding boxes. Since the query selection process is shared between the teacher and student, the generated proposals are identical. During the decoding process, the student uses chunked text tokens$_{1:t-1}$ along with the corresponding proposals, as in the teacher model, to produce corresponding predictions. From the visualization, we can observe that the student's predictions correspond closely to those of the teacher. This ensures that response distillation transfers valid information rather than noise, which is why we term it Correspondence Response Distillation (CRD). Additionally, with corresponding semantic queries and text tokens, the student's object and text prototypes estimated in this mini-batch are more consistent with the teacher's, enhancing the effectiveness of Correspondence Topology Distillation (CTD).

Discussion

Vision-language detectors are emerging as a new trend in the object detection field, yet relevant research remains limited. In this paper, we focus on incremental learning for vision-language detectors. We observe that the semantic structure of these detectors tends to collapse during the incremental learning process, which we identify as a key factor in overcoming catastrophic forgetting. Vision-language representations are highly scalable, with the pretraining process of

VLMs often involving contrastive learning over vast datasets and numerous categories within the same embedding space. Drawing inspiration from this pretraining process, we believe it is crucial to ensure interaction between old and new knowledge to fully exploit the potential of vision-language detectors in incremental object detection (IOD).

Limited performance of new categories. Though our method managed to achieve a better stability-plasticity trade-off, the performance gap to the upper bound for new classes still remains to be explored in the future work.

Absence of Old Objects in the Current Task. Our method mainly focus on IOD setting that assumes old objects exists in future tasks. In this scenario, the knowledge interaction across phases will be impracticable and the teacher model can't provide effective supervision for preserving old knowledge, leading to semantic structure collapse. To address this challenge, a feasible solution is exemplar replay [1,2] which provides old samples extracted from previous tasks. For example, replaying raw images of previous tasks as(Liu et al. 2023b), augmenting current samples with stored old objects as (Liu et al. 2023a).

Significant Domain Gaps Causing Modality Misalignment. The teacher model struggles to capture old knowledge from current task and the modality gap makes it hard to integrate knowledge, resulting in poor performance. For instance, when transferring a model trained on natural images to remote sensing tasks, there is often a stark difference in textual descriptions(e.g., geographic info vs. object names) and visual representations, leading to alignment issues. In such scenarios, freezing original model and using parameter expansion to learn residual knowledge, as suggested in (Jia et al. 2024; Deng et al. 2024), is a practical solution.

References

Deng, J.; Zhang, H.; Ding, K.; Hu, J.; Zhang, X.; and Wang, Y. 2024. Zero-shot Generalizable Incremental Learning for Vision-Language Object Detection. *arXiv preprint arXiv:2403.01680*.

Feng, T.; Wang, M.; and Yuan, H. 2022. Overcoming catastrophic forgetting in incremental object detection via elastic response distillation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, 9427–9436.

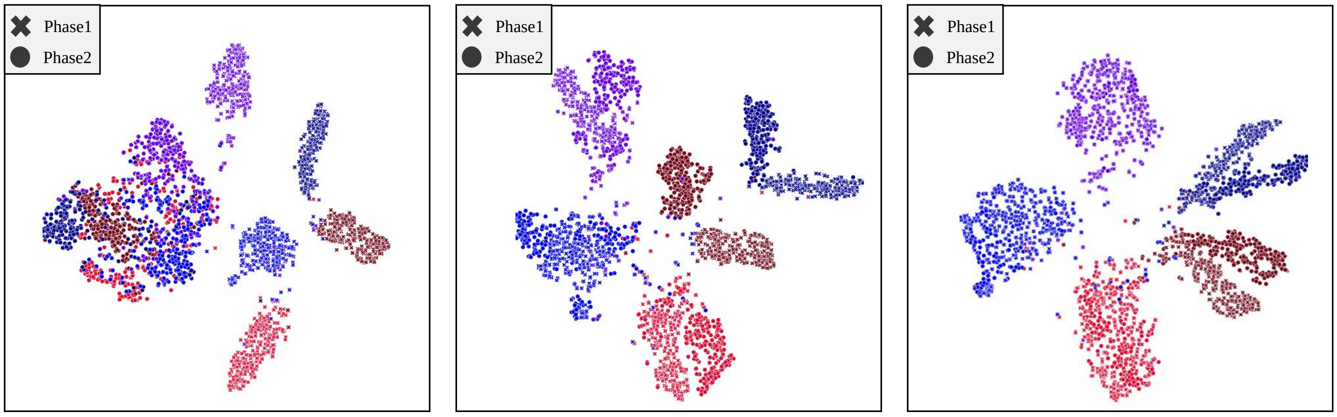
Jia, S.; Wu, T.; Fang, Y.; Zeng, T.; Zhang, G.; and Li, Z. 2024. Purified Distillation: Bridging Domain Shift and Category Gap in Incremental Object Detection. In *Proceedings of the 32nd ACM International Conference on Multimedia*, 1197–1205.

Li, Z.; and Hoiem, D. 2017. Learning without forgetting. *IEEE transactions on pattern analysis and machine intelligence*, 40(12): 2935–2947.

Liu, Y.; Cong, Y.; Goswami, D.; Liu, X.; and van de Weijer, J. 2023a. Augmented box replay: Overcoming foreground shift for incremental object detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 11367–11377.

Liu, Y.; Schiele, B.; Vedaldi, A.; and Rupprecht, C. 2023b. Continual detection transformer for incremental object detection. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 23799–23808.

Van der Maaten, L.; and Hinton, G. 2008. Visualizing data using t-SNE. *Journal of machine learning research*, 9(11).



(a) Local Alignment (b) Ours (w/o CTD) (c) Ours (w/ CTD)

Figure 3: Visualization of semantic query features of old categories under the 70 + 10 setting. Semantic query features from both phases are projected into the same space to directly reflect the changes during the incremental process, where " × " represents semantic query features from phase 1 and " • " represents semantic query features from phase 2. Here, (a) indicates direct fine-tuning with only new text, which serves as a lower bound; (b) represents our method excluding CTD; and (c) represents our complete method.

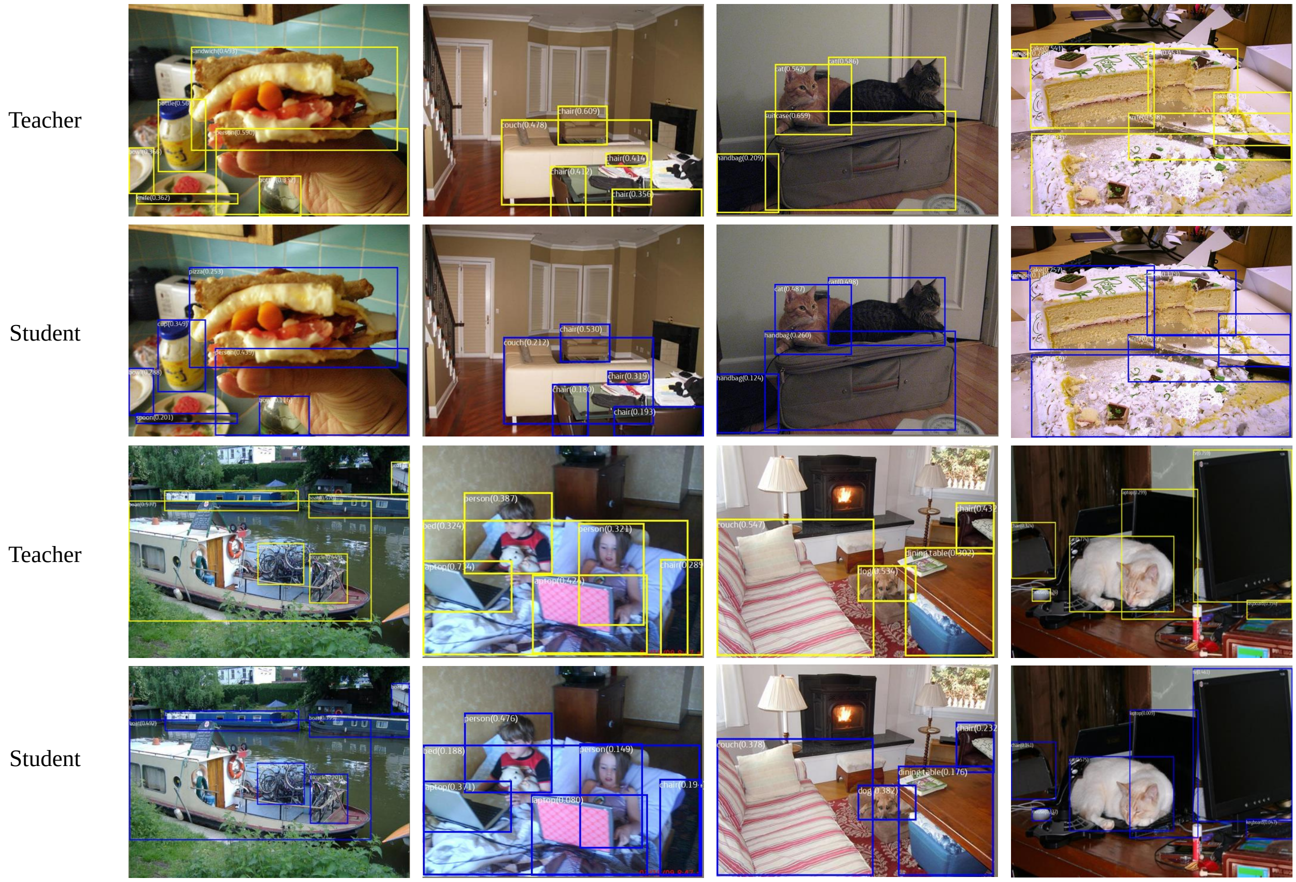


Figure 4: Visualization of Semantic Correspondence Mechanism(SCM), the shown results are extracted from last layer predictions of teacher and student in phase 2 of 70 + 10 setting. For clarity, we only display a subset of bounding boxes and corresponding classification results, where bbox(yellow) represents teacher's predictions and bbox(blue) represents student's predictions. With the introducing of SCM, teacher and student could output corresponding final predictions for distillation.

Algorithm 1: GCD(the t-th phase)

```
**Input:** New dataset $D_t$, New text $prompts_{1:t}$, Old text $prompts_{1:t-1}$, teacher encoder $f_{t-1}$, student encoder $f_t$, teacher decoder $F_{t-1}$, student decoder $F_t$.
**Define:** N: number of object queries
**Define:** L: length of global text token
**Define:** $\phi$ : query selection
**Define:** K: number of decoder layers
**Define:** $\tau$: temperature factor
**for** $epochs$ **do**
    **for** $mini-batch(x, y^{gt}) \in D_t$ **do**
        // **Teacher forward**
        $f_{t-1}(x; prompts_{1:t-1}) \to V^{old}, W^{old}$    // $W^{old} = \{w_1, w_2, \ldots, w_{L-1}\}$
        $\phi(V^{old}, W^{old}) \to O^{old}$    // initialize object queries
        $F_{t-1}(O^{old}, V^{old}, W^{old}) \to Q^{old} \to \hat{y}^{old}$    // generate semantic queries and output predictions $\hat{y}^{old} = \{\hat{s}^{old}, \hat{b}^{old}\}$
        // **Student forward**
        $f_t(x; prompts_{1:t}) \to V, W^{global}$    // obtain global text token $W^{global} = \{w_1, w_2, \ldots, w_L\}$
        $\phi(V, W^{global}) \to O^{global}$    // initialize global object queries
        $F_t(O^{global}, V, W^{global}) \to Q^{global} \to \hat{y}^{global}$    // $\hat{y}^{global} = \{\hat{s}^{gloabal}, \hat{b}^{global}\}$
        // **Global pipeline(Global Alignment):**
        Applying threshold selection on $\hat{s}^{old}$ to generate pseudo labels $\hat{y}^{pseudo}$
        Merge pseudo and GT: $y^{gt} \oplus y^{pseudo} \to y$
        Get $\hat{\sigma}$ by matching $y$ to $\hat{y}^{global}$
        Calculate $\mathcal{L}_{\text{detr}}(\hat{y}^{global}, y)$
        // **Semantic Correspondence Mechanism(SCM):**
        chunk $\{w_1, w_2, \ldots, w_L\} \to \{w_1, w_2, \ldots, w_{L-1}\}$    // obtain local text token $W^{local} = \{w_1, w_2, \ldots, w_{L-1}\}$
        $F_t(O^{old}, V, W^{local}) \to Q^{local} \to \hat{y}^{local}$    // decoding with $O^{old}$ and $W^{local}$to produce corresponding predictions
        // **Local pipeline(CRD):**
        $\mathcal{P}_i^{old} = \text{SoftMax}(\hat{s}_i^{old}/\tau)$    // transform teacher's logits to probabilities, similarly for student $\mathcal{P}_i^{local}$
        $\mathcal{L}_{CRD\_align} = \sum_{i=1}^{N} \alpha_i \mathcal{L}_{KL}(\mathcal{P}_i^{old}, \mathcal{P}_i^{local})$    // $\alpha_i = \max_{c \in C_{1:t-1}}(\hat{s}_i^{old}(c))$ means confidence of teacher's prediction
        $\mathcal{L}_{CRD\_reg} = \sum_{i=1}^{N} \alpha_i \mathcal{L}_{reg}(\hat{b}_i^{old}, \hat{b}_i^{local})$
        $\mathcal{L}_{CRD} = \sum_{k=1}^{K} \gamma \mathcal{L}_{CRD\_align}^{k} + \mathcal{L}_{CRD\_reg}^{k}$    // $\gamma$ is the coefficient for CRD's alignment component
        //**Local pipeline(CTD):**
        $p_c^{local} = \frac{1}{N_c} \sum_{i=1}^{N_c} \alpha_i q_i^{local}$    // obtain object prototype, where $q_i^{local} \in Q^{local}$
        $\hat{p}_c^{local} = \frac{1}{N_b} \sum_{i=1}^{N_b} w_i^{local}$    // obtain text prototype, where $w_i^{local} \in W^{local}$
        $R_{ij}^{local} = \left\| p_i^{local} - p_j^{local} \right\|_2, \quad i, j \in C_{1:t-1}$ // object distance martix, $i, j$ refer to two different classes within $C_{1:t-1}$
        $\hat{R}_{ij}^{local} = \left\| \hat{p}_i^{local} - \hat{p}_j^{local} \right\|_2, \quad i, j \in C_{1:t-1}$    // text distance martix
        $\mathcal{L}_{CTD} = \lambda_1 \left\| R^{local} - R^{old} \right\|_2 + \lambda_2 \left\| \hat{R}^{local} - \hat{R}^{old} \right\|_2$ // $\lambda_1, \lambda_2$ are the coefficients for object and text topology loss.
    **end for**
    $\mathcal{L} = \mathcal{L}_{\text{detr}} + \mathcal{L}_{CRD} + \mathcal{L}_{CTD}$
    update student model via a gradient step
**end for**
```