THE FALSE-NEGATIVE TRAP: STABLE SOURCE-FREE INSTANCE ADAPTATION WITH RECALL-PRIOR MASKING

Anonymous authors
Paper under double-blind review

ABSTRACT

Source-free unsupervised domain adaptation for instance-based detection and segmentation relies heavily on self-training with pseudo-labels. Because target labels are unavailable under the unsupervised setting, a deployable method must deliver strong *terminal* performance under a fixed training schedule rather than relying on peak checkpoint selection. However, in practice, self-training is often unstable: noisy pseudo-labels can accumulate harmful supervision and degrade terminal performance. We study a detector-specific failure mode of this paradigm: when a real target object is omitted from the pseudo-label set, the detector matcher can convert that omission into explicit background supervision. We refer to this mechanism as the *false-negative trap*. We introduce **Recall-Prior Masking** (RPM), a conservative strategy that uses the frozen teacher's region proposal network (RPN) as a class-agnostic recall prior. Rather than converting uncertain proposals into additional pseudo-positive supervision, RPM treats them only as evidence that background supervision may be unsafe. By suppressing possibly unlabelled visual evidence via Informed Cutout and masking localized background losses, RPM aims to prevent harmful optimization. Under a predeclared common-budget protocol on three driving-scene adaptation benchmarks, RPM improves terminal object-detection and instance-segmentation performance and reduces late-training degradation. In a strictly matched comparison with ADABN + FIXED SF-FM, adding RPM improves LAST AP_{50} by $+8.9/+11.6$ segm/bbox on SYNTHIA→CITYSCAPES, $+4.8/+4.8$ on CITYSCAPES→FOGGY CITYSCAPES, and $+2.2/+3.1$ on KITTI→CITYSCAPES. These results show that explicitly protecting potentially omitted objects from erroneous background supervision substantially improves terminal self-training performance and reduces late-training degradation.

1 INTRODUCTION

Self-training with pseudo-labels is a standard strategy for unsupervised domain adaptation (UDA) (Sohn et al., 2020). In source-free UDA, where source data are inaccessible during adaptation, the quality of target pseudo-supervision becomes central for successful transfer (Liang et al., 2020; Yu et al., 2023). This formulation has been widely studied in image classification (Hou & Zheng, 2021; Xia et al., 2024), object detection (Li et al., 2021; 2022a; Liu et al., 2023; Huang et al., 2021; VS et al., 2023; Yoon et al., 2024; Hao et al., 2024), and semantic segmentation (Liu et al., 2021b; Bateson et al., 2022; You et al., 2021). Yet, an operational challenge persists: target labels are unavailable not only during training, but also for checkpoint selection. A deployable method must therefore deliver strong terminal performance at the end of a fixed budget rather than rely on oracle intermediate checkpoint selection. We define this property as *stability*: maintaining high terminal (LAST) performance, avoiding late-training collapse, and exhibiting a minimal BEST→LAST gap (Δ) under a fixed schedule.

A central source of instability is pseudo-label noise, but not all pseudo-label errors affect detector training in the same way. *False positive* pseudo-labels can introduce spurious objects or incorrect class, box, and mask targets, and a broad literature mitigates such errors through confidence filtering, refinement, consistency, ensembling, or weighting. A *false negative* is structurally different. When a real target object is omitted from the pseudo-label set, proposals or predictions covering that object may be assigned to background by the detector's matching procedure. The object is therefore not merely left unsupervised: its omission can be converted into repeated negative supervision.

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.

Over training, this signal can suppress predictions on real target objects and contribute to late-stage degradation. We call this omission-to-negative mechanism the *false-negative trap*.

In two-stage detectors, the region proposal network (RPN) is naturally more recall-oriented than the final prediction head under domain shift. This suggests that the proposal stage can serve as a useful recall prior for identifying regions that may contain real objects omitted from the final pseudo-labels set. Prior work such as LPLD (Yoon et al., 2024) exploits RPN proposals that may recover objects absent from high-confidence detections and transfers their refined semantic information as additional positive supervision. However, under severe domain shift, assigning class semantics to uncertain proposals is prone to secondary FP corruption. Leveraging the RPN high-recall capabilities, we adopt a different, more conservative objective: rather than converting uncertain proposals into positive semantic classes, we exploit proposals solely to prevent assigning background supervision in regions that may contain unlabelled objects.

Based on this principle, we introduce *Recall-Prior Masking* (RPM). We build on the fixed-pseudo-label protocol of Hao et al. (2024) where a source-trained detector is adapted to the target domain using ADABN (Li et al., 2018), after which the adapted model is frozen. This inference-only teacher generates the fixed pseudo-label bank and provides class-agnostic RPN proposals; it receives no gradients, EMA updates, or feedback from the student. A trainable copy initialized from the same adapted model is then optimized on the fixed pseudo-supervision. RPM uses the teacher's RPN proposals as a recall prior and intervenes in two complementary ways. *Informed Cutout* suppresses visual evidence of possibly unlabelled objects, while *Gradient Masking* suppresses background losses in high-recall regions not sufficiently covered by an existing pseudo-label. Crucially, RPM's role is conservative: it only prevents uncertain object-like regions from being confidently learned as background.

We evaluate RPM on three domain shifts spanning synthetic-to-real, adverse-weather, and cross-camera adaptation under a predeclared 40k-iteration budget. This fixed-budget protocol does not assert that it represents the optimal stopping point for every method; rather, it reflects deployable execution without target-label checkpoint selection. Our primary controlled baseline is ADABN + FIXED SF-FM from Hao et al. (2024), which shares the identical protocol as our method, differing solely in the inclusion of RPM. In this strictly matched setting, RPM improves LAST AP_{50} by $+8.9$/$+11.6$ segm/bbox on SYNTHIA→CITYSCAPES, $+4.8$/$+4.8$ on CITYSCAPES→FOGGY CITYSCAPES, and $+2.2$/$+3.1$ on KITTI→CITYSCAPES. Across these shifts, the terminal gains are accompanied by substantially reduced late-training degradation, supporting direct control of omission-induced negative supervision as an effective mechanism for stable source-free instance adaptation.

Contributions. **(C1)** We argue that, in unsupervised adaptation, practical evaluation should prioritize *terminal* performance under a fixed training schedule, as target labels are unavailable for checkpoint selection at deployment time. From this perspective, we isolate the *false-negative trap*, with missed pseudo-label objects (FN) as an important failure mode, contributing to self-training instability. **(C2)** We introduce *Recall-Prior Masking* (RPM), which uses a high-recall, class-agnostic RPN prior to address *false negatives* errors by removing regions containing potential omitted objects from the student supervision. **(C3)** Through strictly matched experiments on three driving-scene adaptation shifts and two instance-level tasks, we show that adding RPM to a fixed-pseudo-label pipeline substantially improves terminal performance and reduces late-training degradation under a fixed adaptation budget. These results show that explicitly handling missed pseudo-label objects is an effective ingredient for making pseudo-label self-training more stable in practice.

2 RELATED WORK

Source-free self-training for object detection. Source-free object detection (SFOD) adapts a source-trained detector to an unlabelled target domain without accessing source data. Existing paradigms explore self-entropy minimization (Li et al., 2021), historical contrastive learning (Huang et al., 2021), style overlook (Li et al., 2022a), instance-relation modeling (VS et al., 2023), and periodic teacher exchange (Liu et al., 2023). Recently, Hao et al. (2024) demonstrated that a competitive baseline can be constructed using comparatively simple components: target statistics adaptation via ADABN (Li et al., 2018), a frozen pseudo-label bank, and weak-strong data augmentations (Sohn et al., 2020). We build upon this fixed-bank methodology, extend it from Faster R-CNN to Mask R-CNN, and address the failure mode that emerges when real target instances are omitted from the pseudo-label bank.

Noisy pseudo-labels and region-aware adaptation. Learning with noisy labels has long distinguished different corruption processes and motivated robust loss correction and sample-selection strategies (Natarajan et al., 2013; Patrini et al., 2017; Han et al., 2018). Detection introduces an additional spatial consequence: an erroneous positive pseudo-label corrupts available positive supervision, whereas an omitted instance is systematically treated as being part of the background. A broad family of pseudo-label methods improves positive-label reliability through confidence filtering, refinement, consistency, or weighting; recent source-free methods also explicitly consider low-confidence or missed target objects. The closest prior work is LPLD (Yoon et al., 2024), which mines low-confidence proposals covering missed objects and distills their refined semantic distributions into the student within an online Mean-Teacher framework. However, LPLD and RPM implement different supervisory principles: LPLD aims to recover uncertain positives by converting proposals into soft foreground signals, whereas RPM has a more conservative approach and simply blocks supervision, preventing adding potential FP-noise from noisy proposals. Other region-aware SFDA methods, such as DPC (Xia et al., 2024), calibrate spatial attention to balance foreground and background cues for image classification, but do not address the spatial anchor-matching step in dense detectors that directly converts omitted instances into background.

Self-training stability and model selection. Cross-domain detectors commonly stabilize self-training using Mean-Teacher exponential moving averages (EMA) (Tarvainen & Valpola, 2017; Liu et al., 2021a; Deng et al., 2021). While EMA mitigates sudden parameter divergence, it remains susceptible to progressive confirmation bias. Conversely, frozen pseudo-label banks (Hao et al., 2024) eliminate online feedback loops but permanently fix label noise, allowing their influence to persist throughout adaptation over long training schedules. However, stability is inseparable from model selection: because target labels are unavailable under the unsupervised assumption, deployable models cannot rely on oracle peak selection. Following the concern raised for source-free evaluation by Hao et al. (2024), we evaluate all reproduced methods under a predetermined adaptation budget, reporting both peak (BEST) and terminal (LAST) performance alongside their gap (Δ) as an operational indicator of training stability.

3 METHODOLOGY

3.1 SOURCE-FREE PSEUDO-LABEL SETUP AND THE FALSE-NEGATIVE TRAP

In source-free UDA, a model f_{θ_S} is first pre-trained on a labelled source dataset $\mathcal{D}_S = \{(x_s, y_s)\}$. During adaptation, the source data become inaccessible, and we adapt using only an unlabelled target training set $\mathcal{D}_T = \{x_t\}$; performance is then evaluated on the target domain. Following Li et al. (2018); Hao et al. (2024), the source model is first adapted to the target domain through a single forward-only ADABN pass over $\mathcal{D}_T$, which refits its BatchNorm statistics. The resulting model $\theta^\star$ is then frozen and used as an inference-only *teacher*. Its final detections form a fixed pseudo-labelled target set $\widehat{\mathcal{D}}_T = \{(x_t, \hat{y}_t)\}$ where $\hat{y}_t$ denotes the pseudo-label set for image x_t. A trainable copy initialized from the same adapted model acts as the *student* and is optimized on $\widehat{\mathcal{D}}_T$, and is therefore supervised by the teacher's predictions. The teacher receives no gradients, EMA updates, or feedback from the student, and the pseudo-label bank remains fixed throughout adaptation.

An erroneous positive pseudo-label corrupts available positive supervision, for example through an incorrect class, box, or mask. Such errors can be mitigated by established confidence filtering, refinement, or weighting strategies. A *false negative* is structurally different: when a real target object is absent from $\hat{y}_t$, the student will treat this region as being background. Thus, omission need not merely remove positive supervision; it can be converted by the detector matcher into explicit negative supervision. We refer to this omission-to-negative pathway as the *false-negative trap*. RPM is designed specifically to intervene on this pathway.

3.2 RECALL-PRIOR MASKING

Class-agnostic recall prior. Two-stage detectors provide a useful asymmetry under domain shift. The region proposal network (RPN) is optimized to identify potentially object-containing regions, whereas the final ROI pipeline additionally requires reliable semantic discrimination. As analyzed in Section 5.1, this recall advantage persists for the frozen ADABN-adapted teacher on the target domain. The teacher cannot reliably label all target objects, but its proposal stage still provides a useful signal about where objects may be present, even when they are absent from the pseudo-label

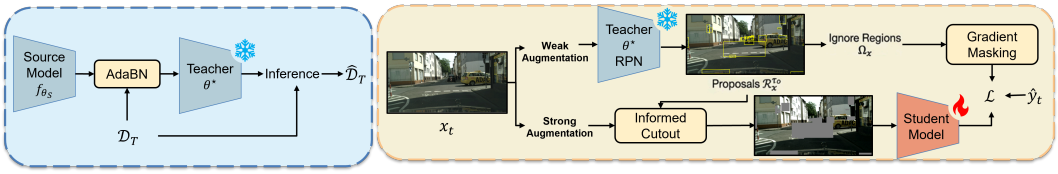


Figure 1: **False-negative trap and Recall-Prior Masking (RPM).** A source-trained detector is adapted with AdaBN and frozen to form the teacher $\theta^\star$, whose final detections define the pseudo-labelled target set $\widehat{\mathcal{D}}_T$. When a real target object is omitted from this bank, student predictions covering it may be matched to background, creating erroneous negative supervision. RPM uses high-recall, class-agnostic teacher-RPN proposals $\mathcal{R}^\star_{x_t}$ as evidence that background supervision may be unsafe. The proposal set guides Informed Cutout, while uncovered proposals define the ignore set Ω_{x_t} used for Gradient Masking.

set. We therefore use RPN objectness as a *recall prior*: a proposal indicates that object presence is plausible, but is not treated as a verified target object and receives no semantic label.

For each target image x_t, we run the frozen backbone and RPN of $\theta^\star$ and retain proposals whose sigmoid objectness exceeds a threshold τ_o, yielding a per-image set

$$\mathcal{R}^\star_{x_t} = \{ r : \mathrm{obj}_{\theta^\star}(r) \geq \tau_o \}. \tag{1}$$

$\mathcal{R}^\star_{x_t}$ is a recall prior, not a precision-tuned detection set: each proposal $r \in \mathcal{R}^\star_{x_t}$ marks a region the teacher considers plausibly object-containing, with no class assignment and no claim that an object actually exists there. RPM uses $\mathcal{R}^\star_{x_t}$ in two complementary ways during the student's training step.

3.2.1 Informed Cutout.

The first intervention acts on the student's input. Let $\tilde{x}_t$ denote the target view used to train the student. For every proposal in the recall-prior set, we replace the corresponding pixels by a constant, following Cutout (DeVries & Taylor, 2017):

$$\tilde{x}_t[\,\text{pixels in } r\,] \leftarrow \kappa, \quad \forall r \in \mathcal{R}^\star_{x_t}. \tag{2}$$

This yields a proposal-guided variant of Cutout in which regions are selected by class-agnostic teacher objectness rather than by random locations. We deliberately use the complete proposal set $\mathcal{R}^\star_{x_t}$, including proposals that overlap existing pseudo-labels. The pseudo-label targets themselves are retained; only the corresponding visual evidence is partially occluded. An alternative variant that protects pseudo-labelled regions from cutout is evaluated in Section 5.2 and performs worse in our experiments. The role of Informed Cutout is not to assert that an unlabelled object is present in every occluded region, but to reduce the student's reliance on visual evidence from regions the teacher itself flags as potentially object-containing, regardless of pseudo-label coverage.

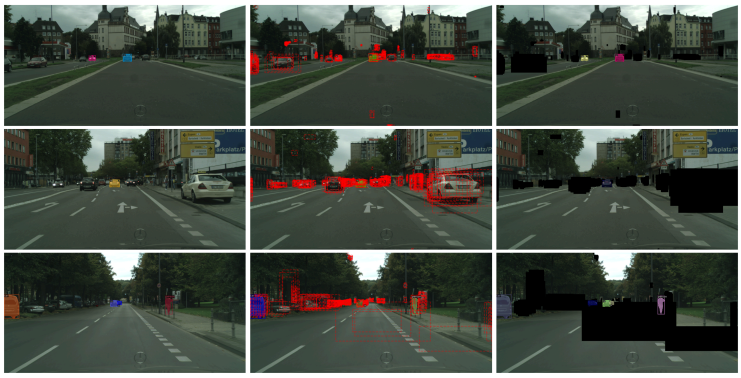


Figure 2: **Qualitative Illustration of Informed Cutout. Left:** Target image with the fixed pseudo-labels, showing many unlabelled objects. **Middle:** High-objectness proposals from the frozen teacher's class-agnostic RPN. Several proposals cover visually plausible object regions that are absent from the pseudo-label set. **Right:** Student input after Informed Cutout. Informed Cutout suppresses the objects' visual evidence from student supervision.

As shown in Figure 2, pseudo-labelled images contain omitted target instances (FN) that standard matching would treat as background. RPM's Informed Cutout leverages high-recall proposals to intercept these regions and suppress harmful unlabelled visual evidence. Because high proposal recall

naturally entails lower precision, attempting to assign semantic classes would introduce secondary FP corruption. Instead, RPM conservatively withhold visual evidence from the student without creating pseudo-positive targets. This separation is central to the formulation: uncertain objectness can make background supervision unsafe without being sufficiently reliable to justify a new positive semantic target.

3.2.2 GRADIENT MASKING.

Input perturbation alone does not remove the false-negative supervision pathway: a student prediction covering an omitted object may still be matched to background within the masked regions. We therefore additionally suppress background supervision in high-recall regions that are not sufficiently covered by an existing pseudo-label. Define

$$\Omega_{x_t} = \left\{ r \in \mathcal{R}^{\star}_{x_t} : \max_{b \in \hat{y}_t} \mathrm{IoU}(r, b) < \eta \right\} \tag{3}$$

as the subset of recall-prior proposals unexplained by the fixed pseudo-label set.

For a matched student prediction j with box b_j and target y_j, we use the binary weight

$$v_j = \begin{cases} 0, & \text{if } y_j = \text{background and } \max\limits_{r \in \Omega_{x_t}} \mathrm{IoU}(b_j, r) \geq \eta, \\ 1, & \text{otherwise,} \end{cases} \tag{4}$$

yielding the student objective

$$\mathcal{L} = \sum_j v_j \, \mathcal{L}_{\mathrm{task}}(p_j, y_j). \tag{5}$$

Thus, background-matched loss contributions are suppressed when the corresponding prediction overlaps the ignore set, while predictions associated with an existing pseudo-label ($y_j \neq$ background) retain their normal positive supervision. The proposal sets used by the two interventions are intentionally different. Informed cutout uses the full proposal set $\mathcal{R}^{\star}_{x_t}$ so the positive target remains present despite input perturbation. Gradient masking uses only the unlabelled subset Ω_{x_t} because zeroing the loss inside a pseudo-labelled region would discard a positive training signal where pseudo-label supervision is available.

The central property of RPM is what it *does not* do. A proposal in $\mathcal{R}^{\star}_{x_t}$ or Ω_{x_t} is never used to add further pseudo-positive objectives. Instead, the proposal is used only to identify regions where a confident background decision may be unsafe. Its role is not to add further pseudo-positive objectives, but to use a recall-oriented prior to identify regions where the student should not be confidently pushed toward naive background supervision.

4 EXPERIMENTS

4.1 SETUP

Benchmarks and metrics. We evaluate on three driving-scene adaptation benchmarks: **(B1)** SYNTHIA→CITYSCAPES (Ros et al., 2016; Cordts et al., 2016), using 6 classes; **(B2)** CITYSCAPES→FOGGY CITYSCAPES (Sakaridis et al., 2018), using 8 classes; **(B3)** KITTI→CITYSCAPES, for which, following prior work (Yoon et al., 2024; Hao et al., 2024), we additionally report Car AP_{50} alongside 8-class mAP_{50}. We evaluate both instance segmentation ($\texttt{segm/AP}_{50}$) and object detection ($\texttt{bbox/AP}_{50}$)

Evaluation protocol. Because target labels are inaccessible for checkpoint selection in unsupervised settings, we evaluate under a fixed budget of 40k iterations, following existing UDA schedules (Hoyer et al., 2022). We treat the terminal checkpoint (LAST) as the primary operational metric, while also reporting oracle peak performance (BEST), peak step ($t^{\star}$), and Δ = BEST − LAST to diagnose stability. The shared endpoint is not intended to represent the individually optimal stopping time of every method; our primary conclusions rely on the strictly matched comparison described below.

Baselines. We extend the source-free OD taxonomy of Hao et al. (2024) to Mask R-CNN, reproducing naive Pseudo-Labeling (SF-PL; Lee, 2013), FixMatch (SF-FM; Sohn et al., 2020), and Unbiased Teacher (SF-UT; Liu et al., 2021a), along with their fixed-bank and ADABN (Li et al., 2018) variants, with optional pseudo-label refinement (Ravi et al., 2024)(†). Our primary controlled comparison is between RPM † and ADABN+FIXED SF-FM†, which only differs in the inclusion of RPM. This comparison therefore isolates the impact of RPM and the effect of suppressing omission-induced

background supervision. External published results using disparate protocols are reported in the appendix for contextual reference.

Implementation. All models use Mask R-CNN with an ImageNet-pretrained ResNet-50 backbone. The fixed pseudo-label bank is generated by the frozen teacher using confidence threshold $\tau = 0.8$, with instance masks refined using the same class-agnostic refinement procedure as the matched baseline. Adaptation runs for 40k iterations using SGD with learning rate 5×10^{-3}, momentum 0.9, weight decay 10^{-4}, batch size 4, on a single NVIDIA V100 GPU. For RPM, we use RPN objectness threshold $\tau_o = 0.5$, proposal exclusion threshold $\eta = 0.3$, and Cutout fill value $\kappa = 127$ (DeVries & Taylor, 2017).

4.2 Stability and Terminal Performance

Tables 1 to 3 report Best, Last, Δ, and $t^\star$ for instance segmentation and object detection on B1–B3. The central result is the strictly matched comparison between RPM and AdaBN+Fixed SF-FM (Simple)†. Because the whole pseudo-label pipeline is unchanged, this comparison isolates the effect of RPM. Adding RPM improves Last by $+8.9$/$+11.6$ AP_{50} (segm/bbox) on B1, $+4.8$/$+4.8$ on B2, and $+2.2$/$+3.1$ on B3. In each case, the terminal gains are larger than the corresponding peak gains, which is consistent with RPM primarily reducing late-training degradation rather than merely shifting the best intermediate checkpoint.

Table 1: **B1: SYNTHIA→Cityscapes.** For each method we report Best (peak iteration), Last (terminal iteration), $\Delta = \text{Best} - \text{Last}$ (stability indicator), and $t^\star$ the peak iteration ($\times 10^3$). Bold marks the best entry in an AP_{50} column and underlining the second-best. Rows marked † apply mask refinement to their pseudo-labels. RPM †, which differs from AdaBN+Fixed SF-FM (Simple)† only in the addition of RPM, shows stability over the full schedule ($\Delta = 0$) and exceeds it by $+8.9$ segm / $+11.6$ bbox AP_{50} at Last.

	segm/AP_{50}				bbox/AP_{50}			
Method	Best	Last	Δ	$t^\star$	Best	Last	Δ	$t^\star$
Source-only	11.0	10.3	0.7	5	18.8	15.4	3.4	5
SF-PL (Lee, 2013; Hao et al., 2024)	0.0	0.0	—	1	0.1	0.0	0.1	1
SF-FM (Sohn et al., 2020; Hao et al., 2024)	0.0	0.0	—	1	0.0	0.0	—	1
SF-UT (Liu et al., 2021a; Hao et al., 2024)	5.3	0.0	5.3	1	7.3	0.0	7.3	1
SF-UT† (Liu et al., 2021a; Hao et al., 2024)	3.0	0.0	3.0	1	4.3	0.0	4.3	1
Fixed SF-PL (Lee, 2013; Hao et al., 2024)	17.2	12.2	5.0	7	26.3	17.7	8.6	3
Fixed SF-PL† (Lee, 2013; Hao et al., 2024)	22.6	16.6	6.0	7	25.9	18.0	7.9	7
Fixed SF-FM (Sohn et al., 2020; Hao et al., 2024)	19.3	14.2	5.1	6	27.2	20.2	7.0	6
AdaBN + Fixed SF-PL (Lee, 2013; Hao et al., 2024)	18.0	16.7	1.3	7	30.1	27.6	2.5	7
AdaBN + Fixed SF-PL† (Lee, 2013; Hao et al., 2024)	28.2	23.6	4.6	11	33.2	27.6	5.6	11
AdaBN + Fixed SF-FM (Sohn et al., 2020; Hao et al., 2024)	22.6	19.1	3.5	2	35.8	<u>29.7</u>	6.1	2
AdaBN + Fixed SF-FM† (Sohn et al., 2020; Hao et al., 2024)	<u>32.7</u>	<u>25.7</u>	7.0	2	<u>39.3</u>	<u>29.7</u>	9.6	2
RPM †	**34.6**	**34.6**	0.0	40	**41.3**	**41.3**	0.0	40

Methods marked † use mask-refined pseudo-labels.

Table 2: **B2: Cityscapes→Foggy Cityscapes.** Same protocol and notation as Table 1. Under the milder weather shift, RPM † exceeds AdaBN+Fixed SF-FM (Simple)† by $+4.8$/$+4.8$ AP_{50} at Last (segm/bbox).

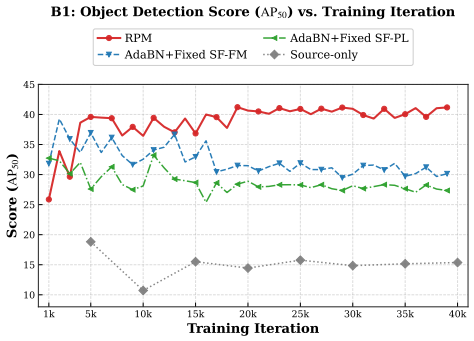
	segm/AP_{50}				bbox/AP_{50}			
Method	Best	Last	Δ	$t^\star$	Best	Last	Δ	$t^\star$
Source-only	22.96	19.92	3.0	10	26.95	22.50	4.5	10
SF-PL (Lee, 2013; Hao et al., 2024)	0.00	0.00	—	1	0.25	0.00	0.3	1
SF-PL† (Lee, 2013; Hao et al., 2024)	4.68	0.00	4.7	1	5.10	0.00	5.1	1
SF-FM (Sohn et al., 2020; Hao et al., 2024)	0.00	0.00	—	1	0.68	0.00	0.7	1
SF-FM† (Sohn et al., 2020; Hao et al., 2024)	10.70	3.66	7.0	1	11.08	4.05	7.0	1
SF-UT (Liu et al., 2021a; Hao et al., 2024)	19.18	0.00	19.2	1	22.16	0.00	22.2	1
SF-UT† (Liu et al., 2021a; Hao et al., 2024)	20.02	0.00	20.0	1	22.26	0.00	22.3	1
AdaBN + Fixed SF-PL (Lee, 2013; Hao et al., 2024)	33.49	32.49	1.0	23	37.75	36.59	1.2	7
AdaBN + Fixed SF-PL† (Lee, 2013; Hao et al., 2024)	33.93	33.10	0.8	3	37.93	36.99	0.9	21
AdaBN + Fixed SF-FM (Sohn et al., 2020; Hao et al., 2024)	37.42	<u>35.41</u>	2.0	10	<u>42.44</u>	<u>39.56</u>	2.9	4
AdaBN + Fixed SF-FM† (Sohn et al., 2020; Hao et al., 2024)	<u>38.35</u>	34.37	4.0	7	41.90	38.58	3.3	7
RPM †	**40.21**	**39.12**	1.1	38	**44.25**	**43.40**	0.9	30

Methods marked † use mask-refined pseudo-labels.

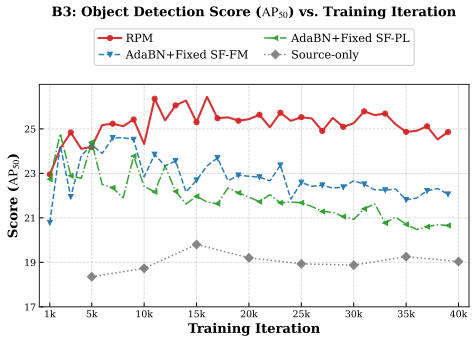
Table 3: **B3: KITTI→CITYSCAPES.** Same protocol and notation as Table 1; we additionally report Car AP_{50} at LAST (column *Car*-L). Under cross-camera shift, RPM attains the highest terminal results on both tasks (+2.2/+3.1 over ADABN+FIXED SF-FM†) and reaches 64.3 Car AP_{50} (vs. 62.0), while preserving much smaller Best→Last gaps than the baselines.

	segm/AP_{50}					bbox/AP_{50}				
Method	BEST	LAST	Δ	$t^\star$	Car-L	BEST	LAST	Δ	$t^\star$	Car-L
Source-only	9.09	8.30	0.8	15	38.06	19.80	19.04	0.8	15	51.34
SF-PL (Lee, 2013; Hao et al., 2024)	0.00	0.00	—	1	0.00	0.13	0.01	0.1	13	0.08
SF-FM (Sohn et al., 2020; Hao et al., 2024)	0.00	0.00	—	1	0.00	0.08	0.01	0.1	1	0.02
SF-FM† (Sohn et al., 2020; Hao et al., 2024)	1.07	0.00	1.1	1	0.00	1.07	0.00	1.1	1	0.00
SF-UT (Liu et al., 2021a; Hao et al., 2024)	7.71	0.00	7.7	1	0.00	19.34	0.00	19.3	1	0.00
AdaBN + Fixed SF-PL (Lee, 2013; Hao et al., 2024)	12.79	11.49	1.3	2	45.97	23.11	19.88	3.2	2	57.74
AdaBN + Fixed SF-PL† (Lee, 2013; Hao et al., 2024)	19.21	17.12	2.1	2	53.50	24.79	21.01	3.8	2	59.49
AdaBN + Fixed SF-FM (Sohn et al., 2020; Hao et al., 2024)	13.50	11.63	1.9	3	47.26	22.79	20.48	2.3	4	59.83
AdaBN + Fixed SF-FM† (Sohn et al., 2020; Hao et al., 2024)	20.11	18.06	2.1	8	54.63	24.61	21.64	3.0	8	61.97
RPM †	**21.29**	**20.22**	1.1	11	**56.74**	**26.44**	**24.71**	1.7	16	**64.33**

Methods marked † use mask-refined pseudo-labels.



(a) Object Detection scores AP_{50} over iterations on B1



(b) Object Detection scores AP_{50} over iterations on B3

RPM consistently improves terminal performance. Across the six benchmark/metric pairs, RPM achieves the strongest LAST performance on all six benchmark/metric pairs. On B1, adding RPM over ADABN+FIXED SF-FM (SIMPLE)† raises LAST from 25.7 to 34.6 segmentation AP_{50} and from 29.7 to 41.3 detection AP_{50}, corresponding to gains of +8.9 and +11.6, respectively. The improvement persists under the milder weather shift B2 (+4.8/+4.8) and the cross-camera shift B3 (+2.2/+3.1). Thus, the effect is not confined to a single source–target pair or to one of the two instance-level tasks.

Stability is preserved across the framework. As noted by Hao et al. (2024), most non-fixed online baselines undergo severe degradation after a few iterations. While the fixed methods show stronger performance and reach the last iteration without collapsing, we can notice that they tend to peak early with $t^\star \leq 11$ on B1, $t^\star \leq 23$ on B2, and $t^\star \leq 8$ on B3, followed by performance degradation until the last iteration. RPM bounds the BEST→LAST gap (Δ) relative to the matched baseline. While the strongest baseline carries a Δ of up to 9.6 on B1, our framework remains remarkably stable. RPM is strictly improving on B1 ($\Delta = 0$) and remains $\Delta \leq 1.7$ everywhere else. This realizes the operational goal of source-free adaptation: under a fixed, unsupervised schedule, RPM supports strong, deployable terminal performance without relying on oracle checkpoint selection by eliminating erroneous background penalties on omitted objects.

Additional comparisons. For completeness, the appendix reports the full per-class results for the reproduced baselines under the same common-budget evaluation, together with comparisons with published SFOD methods. Because those methods differ in training dynamics and in architecture or evaluation protocol, we treat them as contextual evidence rather than as the basis of our primary causal claim. The appendix additionally examines whether adding stronger FP-noise pseudo-label refinement over RPM might improve self-training adaptation in Appendix B.

4.3 TRAINING DYNAMICS AND STABILITY

Section 4.3 shows how performance evolves during adaptation over a 40k iteration schedule rather than only at the terminal checkpoint. Both FIXED SF-PL and FIXED SF-FM reach their strongest

performance early and subsequently degrades. Adding RPM over FIXED SF-FM allows the student to retain considerably stronger performance toward the end of the common budget. On B1, RPM reaches its highest reported score at the final 40k checkpoint, suggesting it could continue to improve over a longer schedule. These trajectories are consistent with the *false-negative trap* motivating RPM; with a fixed pseudo-label bank, an omitted object remains omitted throughout adaptation, so predictions covering that object may repeatedly receive erroneous background supervision. RPM perturbs visual evidence and suppresses background supervision in high-recall regions that may contain omitted objects, reducing a persistent source of harmful supervision.

5 ANALYSIS AND ABLATIONS

We next examine the empirical basis of the recall prior and isolate the principal design choices within RPM.

5.1 EMPIRICAL BASIS OF THE RECALL PRIOR

RPN–ROI recall asymmetry. On Figure 3, we compare the ADABN-adapted teacher's RPN and final ROI pipeline under identical class-agnostic IoU≥ 0.5 matching while sweeping $\tau \in \{0.05, 0.5, 0.7, 0.8, 0.9, 0.95\}$ on B1. Across this range, RPN recall exceeds ROI recall by approximately 19–38 percentage points. At the default RPM operating point $\tau_o = 0.5$, the RPN recalls 71.4% of target objects, compared with 43.0% for the final ROI pipeline. This gap supports the use of RPN objectness as a higher-recall, class-agnostic prior under this shift; it does not require individual proposals to be valid detections or to provide reliable semantic labels.

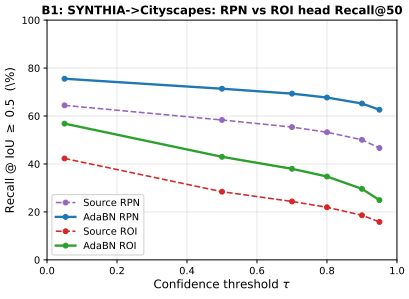


Figure 3: **RPN–ROI recall asymmetry on B1.** Under identical IoU ≥ 0.5 matching, the AdaBN-adapted teacher's RPN retains much higher recall than the full ROI pipeline. At $\tau_o = 0.5$, recall is 71.4% vs. 43.0%.

5.2 RPM DESIGN ABLATIONS

We next isolate the two components of RPM, test whether pseudo-labelled regions should be protected from Informed Cutout, and evaluate sensitivity to the RPN objectness threshold. These experiments use the same fixed pseudo-label bank for every row; hence the pseudo-label supervision is held constant and only the RPM design is varied, isolating changes to the RPM design. The construction of this alternative bank is detailed in the appendix.

Table 4: **RPM design ablations on B1 (SYNTHIA→CITYSCAPES).** All rows use the same pseudo-label configuration. Block (a) decomposes RPM into Gradient Masking (**GM**) and Informed Cutout (**IC**); (b) compares whether pseudo-labelled regions are excluded from IC; (c) varies the RPN objectness threshold τ_o. Both RPM components improve terminal performance, the unprotected IC design performs best among the tested protection strategies, and performance is relatively insensitive to τ_o over the tested B1 range.

Configuration	segm-BEST	segm-LAST	bbox-BEST	bbox-LAST
RPM (IC+GM, unprotected, $\tau_o = 0.5$)	**36.05**	**33.23**	42.37	**38.99**
(a): RPM component decomposition (default = IC+GM)				
IC only	34.99	31.92	41.83	37.59
GM only	31.79	28.89	36.37	33.46
no RPM	30.65	25.89	35.85	29.95
(b): cutout protection (default = unprotected)				
protected (bounding box)	33.49	30.45	39.93	35.35
protected (mask)	33.27	29.93	38.61	34.31
(c): τ_o sensitivity (default $= 0.5$)				
$\tau_o = 0.1$	34.01	32.59	40.94	38.03
$\tau_o = 0.2$	35.47	<u>33.55</u>	**42.62**	<u>39.52</u>
$\tau_o = 0.3$	35.52	33.14	<u>42.76</u>	38.84

RPM Component Decomposition. Table 4a shows that both components improve terminal performance. For bbox AP_{50}, Gradient Masking raises the non-RPM baseline LAST from 29.95 to 33.46, Informed Cutout to 37.59 and their combination to 38.99. The two components therefore play

complementary roles. GM is the intervention that directly suppresses background losses in potentially omitted-object regions, whereas IC provides a stronger proposal-guided input perturbation; their combination performs better than either component in isolation.

Cutout Protection Strategies. Table 4b shows that using the full proposal set for Informed Cutout outperforms explicitly protecting pseudo-labelled boxes or masks, by up to 3.30 `segm` and 4.68 `bbox` AP_{50} at LAST. This supports retaining the full class-agnostic recall prior for input perturbation rather than excluding regions already covered by pseudo-labels.

Threshold Sensitivity (τ_o). As shown in Table 4c, terminal performance varies little over $\tau_o \in \{0.1, 0.2, 0.3, 0.5\}$: `segm` AP_{50} ranges from 32.59 to 33.55 and `bbox` AP_{50} from 38.03 to 39.52, suggesting that RPM is not narrowly threshold-sensitive. This shows that Ω_x functions robustly as a broad recall prior rather than a precision-sensitive detection set; because false positives inside Ω_x merely contribute zero gradient, the student model is safely insulated against over-masking.

6 DISCUSSION

RPM for false-negative supervision. Dense instance adaptation introduces a detector-specific failure mode: when a real target object is omitted from the pseudo-label set, the matching procedure can convert that omission into explicit background supervision. RPM targets this omission-to-negative pathway directly. Rather than assigning uncertain RPN proposals additional positive targets, it uses class-agnostic objectness only to identify regions where a confident background decision may be unsafe. Informed Cutout perturbs visual evidence in proposal-selected regions, while Gradient Masking explicitly suppresses background losses in uncovered high-recall regions. Across the three evaluated domain shifts, adding RPM to a fixed-pseudo-label pipeline improves terminal performance for all six benchmark/task combinations and substantially reduces late-training degradation. This suggests that protecting a model from learning regions where unlabelled objects might be present is a particularly important source of terminal robustness. Additional experiments show that while stronger FP-noise pseudo-label refinement can provide benefits in selected settings, targeting FN-noise remains the main source of improvements for stable self-training with pseudo-labels.

Limitations and scope. Our conclusions are bounded by both architecture and evaluation setting. The teacher adaptation stage relies on ADABN and therefore assumes BatchNorm, while RPM currently uses the explicit RPN of a two-stage detector as its class-agnostic recall prior. Other normalization schemes and detector families would require alternative target-alignment or high-recall objectness signals. Moreover, all experiments use Mask R-CNN on three driving-scene shifts spanning synthetic-to-real, clear-to-foggy, and cross-camera adaptation. The results therefore establish consistency within these evaluated source-free instance-adaptation settings rather than universal generalization; extending the approach to other detector families and domains remains future work.

Terminal evaluation without target-label model selection. Our experiments also highlight the distinction between oracle peak performance and the model obtained at the end of an unsupervised adaptation run. Because labelled target data are unavailable for checkpoint selection, BEST is useful as a diagnostic but is not directly available to the deployment procedure. We therefore report BEST, LAST, the peak iteration $t^\star$, and $\Delta = \text{BEST} - \text{LAST}$ under a predetermined adaptation budget. This common-budget protocol does not assert that a shared schedule is uniformly optimal for all architectures, but answers the operational question of which model is obtained without target supervision. Importantly, adding RPM to a fixed pseudo-label method consistently improves terminal performance and reduces late-training degradation without relying on target-label checkpoint selection.

REPRODUCIBILITY STATEMENT

All experiments use standard public benchmarks (SYNTHIA, CITYSCAPES, and KITTI) and the open-source Detectron2 framework. The experimental setup and optimization schedule are described in Section 4.1; implementation details for the core RPM method are provided in Appendix A, and the additional FP-side pseudo-label refinements are documented separately in Appendix B.

AI USE STATEMENT

Generative AI tools were used solely for sentence-level proofreading, grammatical polishing, and formatting assistance during manuscript preparation. All core research hypotheses, experimental evaluations, methodological implementations, and analytical conclusions were conceived, executed, and validated entirely by the authors.

REFERENCES

Mathilde Bateson, Hoel Kervadec, Jose Dolz, Hervé Lombaert, and Ismail Ben Ayed. Source-free domain adaptation for image segmentation. *Medical Image Analysis*, 82:102617, 2022.

Chaoqi Chen, Zebiao Zheng, Xinghao Ding, Yue Huang, and Qi Dou. Harmonizing transferability and discriminability for adapting object detectors. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8866–8875. IEEE, 2020.

Meilin Chen, Weijie Chen, Shicai Yang, Jie Song, Xinchao Wang, Lei Zhang, Yunfeng Yan, Donglian Qi, Yueting Zhuang, Di Xie, and Shiliang Pu. Learning domain adaptive object detection with probabilistic teacher. In *ICML*, 2022.

Yuhua Chen, Wen Li, Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Domain adaptive Faster R-CNN for object detection in the wild. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 3339–3348, 2018.

Zhihong Chen, Zilei Wang, and Yixin Zhang. Exploiting low-confidence pseudo-labels for source-free object detection. In *ACM MM*, 2023.

Qiaosong Chu, Shuyan Li, Guangyi Chen, Kai Li, and Xiu Li. Adversarial alignment for source free object detection. In *AAAI*, 2023.

Marius Cordts, Mohamed Omran, Sebastian Ramos, Timo Rehfeld, Markus Enzweiler, Rodrigo Benenson, Uwe Franke, Stefan Roth, and Bernt Schiele. The Cityscapes dataset for semantic urban scene understanding. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 3213–3223, 2016.

Jinhong Deng, Wen Li, Yuhua Chen, and Lixin Duan. Unbiased mean teacher for cross-domain object detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 4091–4101, 2021.

Terrance DeVries and Graham W. Taylor. Improved regularization of convolutional neural networks with cutout. In *arXiv preprint arXiv:1708.04552*, 2017.

Bo Han, Quanming Yao, Xingrui Yu, Gang Niu, Miao Xu, Weihua Hu, Ivor Tsang, and Masashi Sugiyama. Co-teaching: Robust training of deep neural networks with extremely noisy labels. *Advances in neural information processing systems*, 31, 2018.

Yan Hao, Florent Forest, and Olga Fink. Simplifying source-free domain adaptation for object detection: Effective self-training strategies and performance insights. In *European Conference on Computer Vision (ECCV)*, 2024.

Zhenwei He and Lei Zhang. Multi-adversarial faster-rcnn for unrestricted object detection. In *ICCV*, 2019.

Zhenwei He and Lei Zhang. Domain adaptive object detection via asymmetric tri-way faster-rcnn. In *ECCV*, 2020.

Yunzhong Hou and Liang Zheng. Source free domain adaptation with image translation, 2021. URL https://arxiv.org/abs/2008.07514.

Lukas Hoyer, Dengxin Dai, and Luc Van Gool. Daformer: Improving network architectures and training strategies for domain-adaptive semantic segmentation. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 9914–9925. IEEE, 2022.

Jiaxing Huang, Dayan Guan, Aoran Xiao, and Shijian Lu. Model adaptation: Historical contrastive learning for unsupervised domain adaptation without source data. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2021.

Mehran Khodabandeh, Arash Vahdat, Mani Ranjbar, and William G Macready. A robust learning approach to domain adaptive object detection. In *ICCV*, 2019.

Dong-Hyun Lee. Pseudo-label: The simple and efficient semi-supervised learning method for deep neural networks. In *ICML Workshop on Challenges in Representation Learning*, 2013.

Congcong Li, Dawei Du, Libo Zhang, Longyin Wen, Tiejian Luo, Yanjun Wu, and Pengfei Zhu. Spatial attention pyramid network for unsupervised domain adaptation. In *European Conference on Computer Vision*, pp. 481–497. Springer, 2020.

Shuaifeng Li, Mao Ye, Xiatian Zhu, Lihua Zhou, and Lin Xiong. Source-free object detection by learning to overlook domain style. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 8014–8023, 2022a.

Xianfeng Li, Weijie Chen, Di Xie, Shicai Yang, Peng Yuan, Shiliang Pu, and Yueting Zhuang. A free lunch for unsupervised domain adaptive object detection without source data. In *Proceedings of the AAAI conference on artificial intelligence*, volume 35, pp. 8474–8481, 2021.

Yanghao Li, Naiyan Wang, Jianping Shi, Xiaodi Hou, and Jiaying Liu. Adaptive batch normalization for practical domain adaptation. *Pattern Recognition*, 80:109–117, 2018.

Yu-Jhe Li, Xiaoliang Dai, Chih-Yao Ma, Yen-Cheng Liu, Kan Chen, Bichen Wu, Zijian He, Kris Kitani, and Peter Vajda. Cross-domain adaptive teacher for object detection. In *CVPR*, 2022b.

Jian Liang, Dapeng Hu, and Jiashi Feng. Do we really need to access the source data? Source hypothesis transfer for unsupervised domain adaptation. In *Int. Conf. on Machine Learning (ICML)*, 2020.

Qipeng Liu, Luojun Lin, Zhifeng Shen, and Zhifeng Yang. Periodically exchange teacher-student for source-free object detection. In *Proc. IEEE Int. Conf. on Computer Vision (ICCV)*, 2023.

Yen-Cheng Liu, Chih-Yao Ma, Zijian He, Chia-Wen Kuo, Kan Chen, Peizhao Zhang, Bichen Wu, Zsolt Kira, and Peter Vajda. Unbiased teacher for semi-supervised object detection. In *Int. Conf. on Learning Representations (ICLR)*, 2021a.

Yuang Liu, Wei Zhang, and Jun Wang. Source-free domain adaptation for semantic segmentation. *arXiv preprint arXiv:2103.16372*, 2021b.

Nagarajan Natarajan, Inderjit S Dhillon, Pradeep K Ravikumar, and Ambuj Tewari. Learning with noisy labels. *Advances in neural information processing systems*, 26, 2013.

Giorgio Patrini, Alessandro Rozza, Aditya Krishna Menon, Richard Nock, and Lizhen Qu. Making deep neural networks robust to label noise: A loss correction approach. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1944–1952, 2017.

Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, et al. SAM 2: Segment anything in images and videos. *arXiv preprint arXiv:2408.00714*, 2024.

German Ros, Laura Sellart, Joanna Materzynska, David Vazquez, and Antonio M. Lopez. The SYNTHIA dataset: A large collection of synthetic images for semantic segmentation of urban scenes. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pp. 3234–3243, 2016.

Kuniaki Saito, Yoshitaka Ushiku, Tatsuya Harada, and Kate Saenko. Strong-weak distribution alignment for adaptive object detection. In *CVPR*, 2019.

Christos Sakaridis, Dengxin Dai, and Luc Van Gool. Semantic foggy scene understanding with synthetic data. *International Journal of Computer Vision*, 126(9):973–992, 2018.

Kihyuk Sohn, David Berthelot, Nicholas Carlini, Zizhao Zhang, Han Zhang, Colin A. Raffel, Ekin D. Cubuk, Alex Kurakin, and Chun-Liang Li. FixMatch: Simplifying semi-supervised learning with consistency and confidence. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2020.

Antti Tarvainen and Harri Valpola. Mean teachers are better role models: Weight-averaged consistency targets improve semi-supervised deep learning results. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2017.

Vibashan VS, Vikram Gupta, Poojan Oza, Vishwanath A Sindagi, and Vishal M Patel. Mega-cda: Memory guided attention for category-aware unsupervised domain adaptive object detection. In *CVPR*, 2021.

Vibashan VS, Poojan Oza, and Vishal M. Patel. Instance relation graph guided source-free domain adaptive object detection. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2023.

Haifeng Xia, Siyu Xia, and Zhengming Ding. Discriminative pattern calibration mechanism for source-free domain adaptation. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 23648–23658. IEEE, 2024.

Lin Xiong, Mao Ye, Dan Zhang, Yan Gan, Xu Li, and Yingying Zhu. Source data-free domain adaptation for a faster r-cnn. In *Pattern Recognition*, 2022.

Ilhoon Yoon, Hyeongjun Kwon, Jin Kim, Junyoung Park, Hyunsung Jang, and Kwanghoon Sohn. Enhancing source-free domain adaptive object detection with low-confidence pseudo label distillation. In *European Conference on Computer Vision*, pp. 337–353. Springer, 2024.

Fuming You, Jingjing Li, Lei Zhu, Zhi Chen, and Zi Huang. Domain adaptive semantic segmentation without source data. In *Proceedings of the 29th ACM international conference on multimedia*, pp. 3293–3302, 2021.

Jingjing Yu et al. A comprehensive survey on source-free domain adaptation. *arXiv preprint arXiv:2302.11803*, 2023.

Siqi Zhang, Lu Zhang, and Zhi Liu. Refined pseudo labeling for source-free domain adaptive object detection. *arXiv preprint arXiv:2303.03728*, 2023.

Ganlong Zhao, Guanbin Li, Ruijia Xu, and Liang Lin. Collaborative training between region proposal localization and classification for domain adaptive object detection. In *European Conference on Computer Vision*, pp. 86–102. Springer, 2020.

A Implementation Details

Teacher and student construction. We first adapt the source-trained Mask R-CNN to the unlabelled target set through a single forward-only AdaBN pass with momentum $m = 0.1$, resetting the BatchNorm running statistics before the sweep. The resulting model $\theta^\star$ is frozen and used as an inference-only teacher. No gradients are computed through the teacher, and it receives no EMA updates or student feedback. A trainable student is initialized from the same AdaBN-adapted model and optimized on the fixed pseudo-labelled target set.

Core fixed pseudo-label bank. All main-paper RPM results use the single-view fixed pseudo-label construction from Hao et al. (2024), employed by the matched baselines. The frozen teacher performs inference once over the target training set, and detections with confidence above $\tau = 0.8$ are retained. The pseudo-label bank is then held fixed throughout adaptation. Where indicated by †, predicted instance masks are refined using the same class-agnostic mask-refinement procedure for both the baseline and RPM.

RPM parameters. For the frozen teacher RPN, we retain proposals with sigmoid objectness $\mathrm{obj}_{\theta^\star}(r) \geq \tau_o$, using $\tau_o = 0.5$ in the default configuration. Informed Cutout fills proposal-selected pixels with the constant $\kappa = 127$. The same values are used across the three main adaptation benchmarks unless otherwise stated. Gradient Masking is applied only to predictions assigned to background that overlap an uncovered recall-prior proposal. Predictions associated with an existing positive pseudo-label retain their standard positive supervision.

Source-model training. All source models use Mask R-CNN with a ResNet-50 backbone initialized from ImageNet weights and trained in Detectron2 on the labelled source training set.

B Additional Analysis of FP-Side Pseudo-Label Refinement

The main paper intentionally evaluates RPM with the standard pseudo-label pipeline in order to isolate the effect of RPM itself. Reliability weighting and multi-view pseudo-label construction are established strategies for improving the quality of positive pseudo-supervision. We therefore treat them here as optional extensions and ask whether stronger positive-pseudo-label refinement is complementary to RPM.

B.1 Multi-view Reliability Construction

For this auxiliary analysis, the frozen teacher processes $K = 4$ views of each target image. Predictions mapped back to a common image coordinate system are grouped through greedy agglomeration (IoU-0.5) over class-compatible candidates, yielding consensus predictions $\mathcal{C}_i^\star$ with consensus class c_i. We compute three reliability cues for each consensus pseudo-instance. The existence reliability is

$$w_{\mathrm{exist}}^{(i)} = \bar{s}_{c_i}^{(i)} \left(\frac{|V^{(i)}|}{K} \right)^2, \tag{6}$$

where $\bar{s}_{c_i}^{(i)}$ is the mean confidence for the consensus class and $|V^{(i)}|$ is the number of views supporting the instance. Spatial reliability is measured from bounding-box agreement,

$$w_{\mathrm{spatial}}^{(i)} = \overline{\mathrm{IoU}}\big(\bar{b}_i, \mathcal{C}_i^\star\big), \tag{7}$$

where $\bar{b}_i$ is the consensus box. Mask reliability is defined from cross-view mask agreement,

$$w_{\mathrm{mask}}^{(i)} = \overline{\mathrm{IoU}}_{\mathrm{mask}}\big(m_i, \mathcal{C}_i^\star \setminus \{m_i\}\big). \tag{8}$$

B.2 Uniform and Decoupled Weighting

We evaluate two ways of applying these reliability cues while keeping RPM unchanged.

RPM + Uniform Weighting. A single pseudo-instance reliability

$$s_i = w_{\mathrm{exist}}^{(i)} w_{\mathrm{spatial}}^{(i)} w_{\mathrm{mask}}^{(i)} \tag{9}$$

weights all supervised heads.

RPM + Decoupled Weighting. The reliability cues are instead routed to the tasks they directly inform:

$$\omega_{\mathrm{cls}}^{(i)} = s_i, \qquad \omega_{\mathrm{loc}}^{(i)} = w_{\mathrm{exist}}^{(i)} w_{\mathrm{spatial}}^{(i)}, \qquad \omega_{\mathrm{msk}}^{(i)} = w_{\mathrm{exist}}^{(i)} w_{\mathrm{mask}}^{(i)}. \tag{10}$$

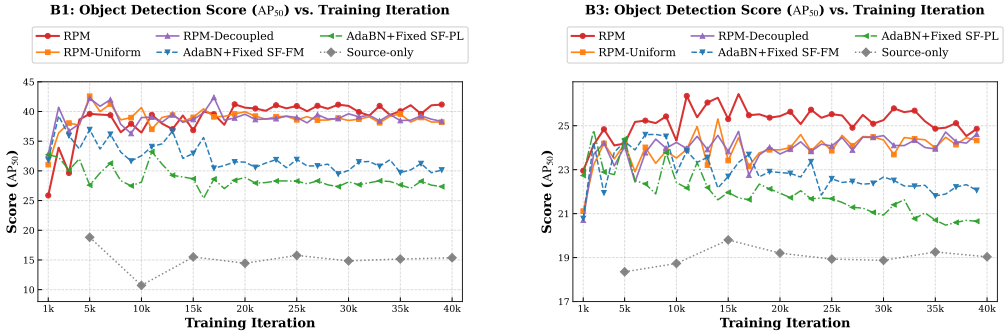
The classification loss is weighted by $\omega_{\mathrm{cls}}^{(i)}$, the localization losses by $\omega_{\mathrm{loc}}^{(i)}$, and the mask loss by $\omega_{\mathrm{msk}}^{(i)}$. Both variants use the same RPM mechanism as the core method; only the pseudo-label construction and positive-supervision weighting differ.

B.3 Training Dynamics with FP-Side Refinement

In Figure 4, the trajectories show that the effect of additional FP-side processing is setting dependent rather than uniformly beneficial. On B1 and B3, the standard single-view RPM configuration retains the strongest terminal result among our variants, while the weighted variants can improve intermediate or class-specific performance. These curves therefore support treating FP-side weighting as a complementary refinement rather than a prerequisite for RPM.

B.4 Effect of FP-Side Pseudo-Label Refinement

We stress-test RPM by varying the pseudo-label construction across four progressively richer designs, holding RPM constant in every row in Section B.4.



(a) Object Detection scores AP_{50} over iterations on B1

(b) Object Detection scores AP_{50} over iterations on B3

Figure 4: **Training trajectories including FP-side refinements.** The core RPM configuration corresponds to the main-paper method. RPM +Uniform and RPM +Decoupled retain the same RPM intervention but use the auxiliary multi-view FP-side pseudo-label refinements defined in Appendix B.1.

Table 5: **Effect of pseudo-label construction with RPM fixed on B1.** RPM is applied identically in every row; only the pseudo-label construction and FP-side weighting strategy vary. All configurations use mask refinement. The standard single-view bank gives the strongest terminal result on B1, whereas richer multi-view weighting improves selected peak metrics.

Pseudo-label strategy	**Score**	**Routing**	segm/AP_{50}		bbox/AP_{50}	
			BEST	LAST	BEST	LAST
Single-view, $\tau = 0.8$ (RPM †)	binary	uniform	34.56	**34.56**	41.33	**41.33**
4-view + binary majority vote	binary	uniform	32.38	30.42	40.18	35.84
4-view + binary unanimous vote	binary	uniform	33.36	28.04	38.84	33.07
4-view + weighting (Uniform)	continuous	uniform	35.53	32.89	**42.57**	38.77
4-view + per-task weight (Decoupled)	continuous	per-task	**36.05**	<u>33.23</u>	42.37	<u>38.99</u>

Holding RPM fixed isolates the effect of changing the positive pseudo-supervision. The stress test confirms that RPM's terminal-iteration dominance is robust to pseudo-label design. With RPM held constant, the simplest single-view bank (RPM) attains the strongest LAST on both metrics ($\Delta = 0$). Binarising multi-view consensus into majority or unanimous votes degrades both BEST and LAST: information lost by binarisation outweighs the benefit of multi-view averaging. Restoring continuous scoring via Uniform-weighting recovers $+1.0$ BEST segm over RPM; per-task routing (Decoupled) adds another $+0.5$. The weighted variants thus lift peak performance at the cost of a small non-zero Δ from richer pseudo-label noise memorisation. The FP-weighted methods show higher peak results, while simply using RPM showhigher terminal results. The consistent main-paper gains therefore do not depend on the FP-refinement mechanism but rather on handling FN with RPM.

C PER-CLASS BREAKDOWNS

B1 segm/bbox at LAST. Tables 6 and 7 detail the full per-class breakdowns for both segmentation and bounding box performance on the B1 benchmark.

B2 per-class results at LAST. Table 8 details the segmentation AP_{50} at LAST on the B2 benchmark. Bounding box results are integrated into our comprehensive cross-architecture comparisons in Table 10 in Section D

B3 per-class results at LAST. Table 9 details the segmentation AP_{50} at LAST on the B3 benchmark. Bounding box results are integrated into our comprehensive cross-architecture comparisons in Table 11 in Section E.

Table 6: **Per-class `segm/AP`$_{50}$ at LAST on B1 (SYNTHIA→CITYSCAPES).** Bold marks the best entry and underlining the second-best. Rows marked † apply mask refinement to their pseudo-labels. The core RPM configuration obtains the highest overall mAP and the best score on four of six classes.

Method	bicycle	bus	car	motorc.	person	rider	**mAP**
Source-only	-	-	-	-	-	-	10.25
SF-PL (Hao et al., 2024)	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SF-FM (Hao et al., 2024)	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SF-UT (Hao et al., 2024)	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SF-UT† (Hao et al., 2024)	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Fixed SF-PL (Hao et al., 2024)	0.00	8.95	26.77	0.99	29.49	7.21	12.24
Fixed SF-PL † (Hao et al., 2024)	10.89	10.57	31.07	0.99	33.36	12.72	16.60
Fixed SF-FM (Hao et al., 2024)	0.00	11.33	30.45	0.99	33.12	9.16	14.18
AdaBN + Fixed SF-PL (Hao et al., 2024)	0.01	8.01	34.05	0.03	38.89	18.94	16.66
AdaBN + Fixed SF-PL † (Hao et al., 2024)	17.54	10.14	39.37	4.30	44.15	26.03	23.59
AdaBN + Fixed SF-FM (Hao et al., 2024)	0.04	11.12	37.11	0.30	42.54	23.20	19.05
AdaBN + Fixed SF-FM † (Hao et al., 2024)	21.06	8.43	43.23	6.13	46.59	28.76	25.70
RPM †	**26.79**	<u>22.21</u>	**53.07**	<u>14.93</u>	**48.23**	**42.11**	**34.56**
RPM + Uniform†	<u>24.91</u>	22.13	<u>49.54</u>	**15.73**	<u>47.55</u>	<u>37.48</u>	32.89
RPM + Decoupled†	24.90	**25.78**	49.35	14.79	47.08	<u>37.48</u>	<u>33.23</u>

Methods marked † use mask-refined pseudo-labels.

Table 7: **Per-class `bbox/AP`$_{50}$ at LAST on B1 (SYNTHIA→CITYSCAPES).** Bold marks the best entry and underlining the second-best. Rows marked † apply mask refinement to their pseudo-labels. RPM achieves the highest LAST mAP and the per-class best on 5 of 6 classes; among the prior baselines, only ADABN-aligned fixed pseudo-label variants retain non-trivial LAST performance.

Method	bicycle	bus	car	motorc.	person	rider	**mAP**
Source-only	-	-	-	-	-	-	15.37
SF-PL (Hao et al., 2024)	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SF-FM (Hao et al., 2024)	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SF-UT (Hao et al., 2024)	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SF-UT† (Hao et al., 2024)	0.00	0.00	0.00	0.00	0.00	0.00	0.00
Fixed SF-PL (Hao et al., 2024)	13.60	10.25	31.36	0.99	35.10	15.13	17.74
Fixed SF-PL † (Hao et al., 2024)	12.70	11.49	33.38	0.99	35.67	14.03	18.04
Fixed SF-FM (Hao et al., 2024)	15.71	14.15	34.87	2.13	38.29	16.07	20.20
AdaBN + Fixed SF-PL (Hao et al., 2024)	22.95	11.48	40.76	5.17	47.81	37.45	27.60
AdaBN + Fixed SF-PL † (Hao et al., 2024)	24.31	12.60	43.14	3.39	48.51	33.75	27.61
AdaBN + Fixed SF-FM (Hao et al., 2024)	25.57	12.62	46.28	5.88	51.65	35.90	29.65
AdaBN + Fixed SF-FM † (Hao et al., 2024)	28.21	8.45	48.22	5.67	51.58	36.28	29.74
RPM †	**37.94**	21.82	**61.73**	**18.21**	**52.82**	**55.48**	**41.33**
RPM + Uniform†	<u>33.89</u>	<u>22.59</u>	<u>56.55</u>	17.81	<u>51.75</u>	<u>50.01</u>	38.77
RPM + Decoupled†	33.78	**26.03**	55.61	<u>18.06</u>	51.73	48.70	<u>38.99</u>

Methods marked † use mask-refined pseudo-labels.

D CONTEXTUAL CROSS-ARCHITECTURE COMPARISON ON B2

Table 10 situates our B2 results within the broader UDAOD and SFOD literature. Prior numbers are reproduced from Hao et al. (2024) and Yoon et al. (2024); all Mask R-CNN ResNet-50 FPN rows are our own measurements under a unified implementation.

Evaluation protocol. A key distinction is the evaluation protocol. REPORTED denote the reported metrics from each paper, following each method's native protocol. As noted by Hao et al. (2024), some UDA papers do not explicitly state the training schedule, hindering possible target-domain evaluation checkpoint selection. We additionally report LAST (iteration 40k), which corresponds to a fixed training schedule with no target-side checkpoint selection and therefore better reflects deployable performance under the unsupervised assumption.

Table 8: **Per-class `segm/AP`$_{50}$ at LAST on B2 (CITYSCAPES→FOGGY CITYSCAPES).** Core RPM improves over the matched ADABN+FIXED SF-FM† baseline on all eight reported classes. The optional Uniform Weighting refinement yields additional gains on *bus*, *train*, and *truck*, producing the highest overall mAP among the auxiliary variants.

Method	bicycle	bus	car	motorc.	person	rider	train	truck	**mAP**
Source-only	21.91	26.03	32.85	11.68	25.98	31.85	2.00	7.07	19.92
SF-PL (Hao et al., 2024)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SF-PL† (Hao et al., 2024)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SF-FM (Hao et al., 2024)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SF-FM† (Hao et al., 2024)	0.00	0.00	29.27	0.00	0.00	0.00	0.00	0.00	3.66
SF-UT (Hao et al., 2024)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SF-UT† (Hao et al., 2024)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
AdaBN + Fixed SF-PL (Hao et al., 2024)	35.06	37.72	44.62	29.49	37.43	42.97	11.43	21.24	32.49
AdaBN + Fixed SF-PL † (Hao et al., 2024)	35.52	38.13	44.66	29.56	38.10	38.63	19.71	20.49	33.10
AdaBN + Fixed SF-FM (Hao et al., 2024)	37.29	41.36	48.85	30.80	40.62	48.41	10.67	25.29	35.41
AdaBN + Fixed SF-FM † (Hao et al., 2024)	37.36	40.09	48.25	30.83	41.20	40.62	10.08	26.55	34.37
RPM †	**41.27**	41.47	**52.88**	**32.93**	**43.21**	**41.47**	31.02	28.68	39.12
RPM + Uniform†	35.92	**46.84**	50.77	32.41	41.01	39.39	**37.33**	**31.39**	**39.38**
RPM + Decoupled†	35.52	44.07	49.80	31.75	40.33	38.30	34.80	30.36	38.12

Methods marked † use mask-refined pseudo-labels.

Table 9: **Per-class `segm/AP`$_{50}$ at LAST on B3 (KITTI→CITYSCAPES).** Core RPM achieves the highest overall mAP and the best score on four of eight classes. The optional Uniform and Decoupled weighting variants improve selected classes but do not improve the overall terminal mAP in this setting. Bus AP is 0 for all adapted variants reported here.

Method	bicycle	bus	car	motorc.	person	rider	train	truck	**mAP**
Source-only	4.70	3.02	38.06	1.81	5.26	1.36	4.95	7.23	8.30
SF-PL (Hao et al., 2024)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SF-FM (Hao et al., 2024)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SF-FM† (Hao et al., 2024)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
SF-UT (Hao et al., 2024)	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00	0.00
AdaBN + Fixed SF-PL (Hao et al., 2024)	9.24	0.00	45.97	3.10	12.81	7.05	2.48	11.29	11.49
AdaBN + Fixed SF-PL † (Hao et al., 2024)	21.87	0.00	53.50	4.16	22.04	18.13	2.48	14.83	17.12
AdaBN + Fixed SF-FM (Hao et al., 2024)	8.88	0.00	47.26	2.34	14.38	8.30	1.65	10.21	11.63
AdaBN + Fixed SF-FM † (Hao et al., 2024)	24.32	0.00	54.63	3.94	25.41	18.41	2.48	15.31	18.06
RPM †	**26.88**	0.00	**56.74**	4.82	28.85	**24.80**	**4.95**	14.69	**20.22**
RPM + Uniform†	26.72	0.00	55.99	6.22	**31.40**	21.78	0.00	**15.72**	19.73
RPM + Decoupled†	26.70	0.00	56.02	**6.88**	30.71	20.37	0.00	15.56	19.53

Methods marked † use mask-refined pseudo-labels.

Cross-architecture setting. Our experiments use Mask R-CNN with a ResNet-50 FPN backbone, whereas much of the prior literature uses Faster R-CNN variants. Our source-only baseline (22.5 mAP) is therefore lower than the stronger Faster R-CNN source-only numbers in the table (e.g., 25.2 for ResNet-50 and 26.9 for VGG16-BN). Gains should be interpreted relative to this weaker starting point rather than as a like-for-like architecture comparison.

Reading the comparison. Within the reproduced Mask R-CNN setting, core RPM improves both peak and terminal detection performance relative to the matched fixed-bank baseline. The optional Uniform Weighting refinement provides a further gain on B2, reaching 45.15 AP_{50} at LAST, while Decoupled Weighting produces a similar terminal result to core RPM. The per-class results in Table 8 show that the Uniform refinement's additional gain is concentrated particularly on *bus*, *train*, and *truck*. The Faster R-CNN values provide broader literature context but are not directly comparable because of architecture and protocol differences. We therefore do not use their numerical ordering as evidence of superiority.

Effect of pseudo-mask refinement. We additionally compare variants with and without mask refinement. Across matched baseline pairs, methods trained on mask-refined pseudo-labels show higher segm/AP50. This dynamic highlights that a significant portion of pseudo-label "noise" is directly attributable to the segmentation head's predictions, particularly under strong cross-domain shifts. It also demonstrates that class-agnostic mask refinement can serve as an effective tool for self-training with pseudo-labels on segmentation tasks facing domain shift. However, because this intervention merely refines mask boundaries, it does not intrinsically improve object detection performance; methods with and without mask refinement perform similarly on the bounding box task, with effects on bbox/AP50 being mixed or sometimes negative. For instance, on B1, Fixed SF-PL gains clearly in segm under SAM2 ($17.2 \rightarrow 22.6$ BEST; $12.2 \rightarrow 16.6$ LAST), yet its bbox BEST slightly decreases ($26.3 \rightarrow 25.9$); AdaBN + Fixed SF-PL leaves bbox LAST unchanged ($27.6 \rightarrow 27.6$). On the B2 contextual comparison, ADABN + FIXED SF-FM achieves 39.56 mAP at LAST, versus 38.58 for its SAM2 variant. Thus, SAM2 alone does not explain stronger detection results. The more important fact for our claim is that our methods still outperform the strongest SAM2-augmented baselines on both segm and bbox, at both BEST and LAST in B1, and at LAST on B2 and B3 as well, while using the same pseudo-label bank. This supports the argument that the main gains come from better training dynamics and false-negative control.

Discussion. These results highlight a central methodological point: *peak performance is not the same as deployable performance*. Methods that exhibit strong performance under checkpoint selection can degrade substantially during continued training. Under a fixed schedule, our method combines strong accuracy with stable convergence, which is exactly the behavior an unsupervised adaptation method should optimize for.

E CONTEXTUAL CROSS-ARCHITECTURE COMPARISON ON B3

Protocol. We extend the same contextual comparison to B3 (KITTI→CITYSCAPES). Following convention in this benchmark (Hao et al., 2024; Yoon et al., 2024), prior detection-only work reports only Car AP_{50}, so per-class columns are blank for those rows; Mask R-CNN methods (ours and our reproductions) report all 8 classes plus mAP. The same protocol distinction (REPORTED vs. LAST) and the same backbone-asymmetry observation carry over from Table 10.

Reading. Within our Mask R-CNN experiments, the core RPM configuration is strongest on B3 both at the oracle peak and at the terminal checkpoint, reaching 26.4 and 24.7 bbox mAP, respectively. The optional Uniform and Decoupled weighting variants reach terminal mAPs of 24.5 and 24.1, indicating that stronger FP-side refinement does not improve the aggregate B3 result. For the benchmark-standard Car category, core RPM reaches 64.3 AP_{50} at LAST, compared with 62.0 for the matched ADABN+FIXED SF-FM† baseline. Published Faster R-CNN values are included only as contextual reference because the detector architecture and evaluation protocols differ.

Table 10: **Contextual comparison on B2 (CITYSCAPES→FOGGY CITYSCAPES), `bbox/AP`$_{50}$.** Bold and underline mark the best and second-best SFOD entries in a column, evaluated independently for the REPORTED and LAST settings. The **Iteration** column makes the protocol distinction explicit: REPORTED denotes the checkpoint of maximum target-test mAP, whereas LAST denotes iteration 40k of a fixed schedule with no target-side checkpoint selection. Only our Mask R-CNN rows report LAST, which is the more operationally meaningful metric under the unsupervised assumption.

Setting	Method	Detector	Backbone	Iteration	Person	Rider	Car	Truck	Bus	Train	Motor	Bicycle	mAP
Source-only	Source-only (Hao et al., 2024)	Faster R-CNN	VGG16	—	20.3	27.2	30.3	12.7	17.6	4.9	28.3	27.3	20.0
	Source-only (Hao et al., 2024)	Faster R-CNN	VGG16-BN	—	31.7	38.5	38.9	18.2	28.4	4.9	23.2	32.0	26.9
	Source-only (Yoon et al., 2024)	Faster R-CNN	ResNet-50	—	29.3	34.1	35.8	15.4	26.0	9.1	22.4	29.7	25.2
	Source-only	Mask R-CNN	ResNet-50	LAST	28.7	36.7	34.1	7.1	26.0	2.0	15.2	30.1	22.5
UDAOD	DA-Faster (Chen et al., 2018)	Faster R-CNN	VGG16	REPORTED	25.0	31.0	40.5	22.1	35.3	20.2	20.0	27.1	27.6
	MeGA-CDA (VS et al., 2021)	Faster R-CNN	VGG16	REPORTED	25.4	52.4	49.0	37.7	46.9	34.5	39.0	49.2	41.8
	UMT (Deng et al., 2021)	Faster R-CNN	VGG16	REPORTED	33.0	46.7	48.6	34.1	56.5	46.8	30.4	37.3	41.7
	PT (Chen et al., 2022)	Faster R-CNN	VGG16	REPORTED	40.2	48.8	59.7	30.7	51.8	30.6	35.4	44.5	42.7
	AT (Li et al., 2022b)	Faster R-CNN	VGG16-BN	REPORTED	56.3	51.9	64.2	38.5	45.5	55.1	54.3	35.0	50.9
SFOD	MT (Tarvainen & Valpola, 2017)	Faster R-CNN	ResNet-50	REPORTED	37.4	43.0	45.0	27.2	37.2	25.1	28.2	38.2	34.3
	SED (Li et al., 2021)	Faster R-CNN	VGG16	REPORTED	32.6	40.4	44.0	21.7	34.3	11.8	25.3	34.5	30.6
	LODS (Li et al., 2022a)	Faster R-CNN	VGG16	REPORTED	34.0	45.7	48.8	27.3	39.7	19.6	33.2	37.8	35.8
	A^{2}SFOD (Chu et al., 2023)	Faster R-CNN	VGG16	REPORTED	32.3	44.1	44.6	28.1	34.3	29.0	31.8	38.9	35.4
	IRG (VS et al., 2023)	Faster R-CNN	ResNet-50	REPORTED	37.4	45.2	51.9	24.4	39.6	25.2	31.5	41.6	37.1
	HCL (Huang et al., 2021)	Faster R-CNN	VGG16	REPORTED	38.7	46.0	47.9	**33.0**	45.7	38.9	32.8	34.9	39.7
	RPL (Zhang et al., 2023)	Faster R-CNN	VGG16	REPORTED	37.9	52.7	52.0	29.9	46.7	21.7	34.5	46.5	40.2
	PETS (Liu et al., 2023)	Faster R-CNN	VGG16	REPORTED	42.0	48.7	56.3	19.3	39.3	5.5	34.2	41.6	35.9
	LPLD (Yoon et al., 2024)	Faster R-CNN	ResNet-50	REPORTED	38.3	42.9	52.5	28.4	42.1	43.9	33.4	41.8	40.4
	LPLD (Yoon et al., 2024)	Faster R-CNN	VGG16	REPORTED	39.7	49.1	56.6	29.6	46.3	26.4	36.1	43.6	40.9
	AdaBN (Li et al., 2018)	Faster R-CNN	VGG16-BN	REPORTED	38.5	44.1	49.9	26.0	39.2	16.6	29.0	37.4	35.1
	AdaBN + Fixed SF-PL (Hao et al., 2024)	Faster R-CNN	VGG16-BN	REPORTED	40.0	45.1	52.3	26.9	43.7	28.2	31.0	41.2	38.5
	AdaBN + Fixed SF-FM (Hao et al., 2024)	Faster R-CNN	VGG16-BN	REPORTED	41.5	48.1	55.1	30.8	49.3	**53.0**	34.2	44.1	44.5
	SF-UT (Hao et al., 2024)	Faster R-CNN	VGG16	REPORTED	36.5	44.6	54.8	30.2	45.3	31.8	29.5	41.0	39.2
	SF-UT (Hao et al., 2024)	Faster R-CNN	VGG16-BN	REPORTED	40.9	48.0	58.9	29.6	**51.9**	50.2	36.2	44.1	45.0
	SF-UT (Hao et al., 2024)	Mask R-CNN	ResNet-50	REPORTED	28.8	36.4	34.8	7.8	22.4	0.6	15.8	30.7	22.2
	SF-UT† (Hao et al., 2024)	Mask R-CNN	ResNet-50	REPORTED	28.5	34.9	34.9	9.7	24.1	1.4	14.9	29.8	22.3
	AdaBN + Fixed SF-PL (Hao et al., 2024)	Mask R-CNN	ResNet-50	REPORTED	42.0	49.1	51.8	23.4	35.9	18.3	32.7	48.9	37.8
	AdaBN + Fixed SF-PL † (Hao et al., 2024)	Mask R-CNN	ResNet-50	REPORTED	42.7	51.3	48.7	21.3	38.7	18.7	35.1	46.9	37.9
	AdaBN + Fixed SF-FM (Hao et al., 2024)	Mask R-CNN	ResNet-50	REPORTED	48.0	**57.4**	58.7	20.9	40.7	26.3	36.8	50.8	42.4
	AdaBN + Fixed SF-FM † (Hao et al., 2024)	Mask R-CNN	ResNet-50	REPORTED	47.2	55.6	57.3	28.9	40.7	20.1	36.4	49.0	41.9
	RPM †	Mask R-CNN	ResNet-50	BEST	48.1	56.4	58.0	28.7	42.2	27.8	39.1	**53.8**	44.3
	RPM + Uniform†	Mask R-CNN	ResNet-50	BEST	**49.6**	54.7	**59.5**	30.0	46.1	33.9	**40.8**	53.1	**46.0**
	RPM + Decoupled†	Mask R-CNN	ResNet-50	BEST	46.1	54.7	55.1	30.1	44.5	35.4	39.2	51.8	44.6
	SF-UT (Hao et al., 2024)	Mask R-CNN	ResNet-50	LAST	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	SF-UT† (Hao et al., 2024)	Mask R-CNN	ResNet-50	LAST	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	AdaBN + Fixed SF-PL (Hao et al., 2024)	Mask R-CNN	ResNet-50	LAST	41.6	50.0	47.9	22.5	37.7	11.4	36.1	45.6	36.6
	AdaBN + Fixed SF-PL † (Hao et al., 2024)	Mask R-CNN	ResNet-50	LAST	41.3	50.3	47.0	20.5	38.1	18.6	34.7	45.5	37.0
	AdaBN + Fixed SF-FM (Hao et al., 2024)	Mask R-CNN	ResNet-50	LAST	45.7	**55.2**	52.4	26.8	41.4	8.7	38.1	48.4	39.6
	AdaBN + Fixed SF-FM † (Hao et al., 2024)	Mask R-CNN	ResNet-50	LAST	44.7	**52.2**	51.4	26.6	40.1	10.1	36.7	47.0	38.6
	RPM †	Mask R-CNN	ResNet-50	LAST	**48.7**	55.1	**57.2**	28.7	41.5	25.0	38.0	**53.1**	43.4
	RPM + Uniform†	Mask R-CNN	ResNet-50	LAST	46.3	53.4	55.1	**32.7**	**46.8**	**37.3**	38.2	51.5	**45.2**
	RPM + Decoupled†	Mask R-CNN	ResNet-50	LAST	45.4	54.7	54.9	29.2	44.6	26.9	**40.4**	51.2	43.4
Oracle	Oracle (Hao et al., 2024)	Faster R-CNN	VGG16	—	51.3	57.5	70.2	30.9	60.5	26.9	40.0	50.4	43.5
	Oracle (Yoon et al., 2024)	Faster R-CNN	ResNet-50	—	38.7	46.9	56.7	35.5	49.4	44.7	35.9	38.8	43.1

Methods marked † use mask-refined pseudo-labels.

Table 11: **Contextual comparison on B3 (KITTI→CITYSCAPES), `bbox/AP`$_{50}$.** Bold and underline mark the best and second-best SFOD entries in a column, evaluated independently for the REPORTED and LAST settings. The **Iteration** column again distinguishes oracle-like checkpoint selection (REPORTED) from strict fixed-schedule evaluation (LAST, iteration 40k). Dashes indicate classes not reported by prior detection-only work.

Setting	Method	Detector	Backbone	Iteration	Person	Rider	Car	Truck	Bus	Train	Motor	Bicycle	mAP
Source-only	Source-only (Hao et al., 2024)	Faster R-CNN	VGG16	—	-	-	14.6	-	-	-	-	-	-
	Source-only (Hao et al., 2024)	Faster R-CNN	VGG16-BN	—	-	-	29.1	-	-	-	-	-	-
	Source-only (Yoon et al., 2024)	Faster R-CNN	ResNet-50	—	-	-	33.9	-	-	-	-	-	-
	Source-only	Mask R-CNN	ResNet-50	LAST	18.4	31.7	51.3	11.7	3.9	5.0	7.8	22.5	19.0
UDAOD	DA-Faster	Faster R-CNN	VGG16	REPORTED	-	-	38.5	-	-	-	-	-	-
	SWDA (Saito et al., 2019)	Faster R-CNN	VGG16	REPORTED	-	-	37.9	-	-	-	-	-	-
	Pseudo-Label (Khodabandeh et al., 2019)	Faster R-CNN	InceptionV2	REPORTED	-	-	40.2	-	-	-	-	-	-
	NL (Khodabandeh et al., 2019)	Faster R-CNN	InceptionV2	REPORTED	-	-	43.0	-	-	-	-	-	-
	ATF (He & Zhang, 2020)	Faster R-CNN	VGG16	REPORTED	-	-	42.1	-	-	-	-	-	-
	HTCN (Chen et al., 2020)	Faster R-CNN	VGG16	REPORTED	-	-	42.5	-	-	-	-	-	-
	SAPNet (Li et al., 2020)	Faster R-CNN	VGG16	REPORTED	-	-	43.4	-	-	-	-	-	-
	CST-DA (Zhao et al., 2020)	Faster R-CNN	VGG16	REPORTED	-	-	43.6	-	-	-	-	-	-
	Cycle DA	Faster R-CNN	VGG16	REPORTED	-	-	41.7	-	-	-	-	-	-
	MeGA-CDA (VS et al., 2021)	Faster R-CNN	VGG16	REPORTED	-	-	43.0	-	-	-	-	-	-
	SGA-S	Faster R-CNN	ResNet-101	REPORTED	-	-	43.5	-	-	-	-	-	-
	MAF (He & Zhang, 2019)	Faster R-CNN	VGG16	REPORTED	-	-	41.0	-	-	-	-	-	-
	Robust DA (Khodabandeh et al., 2019)	Faster R-CNN	ResNet-50	REPORTED	-	-	42.9	-	-	-	-	-	-
	Unbiased DA (Deng et al., 2021)	Faster R-CNN	VGG16	REPORTED	-	-	43.1	-	-	-	-	-	-
	PT (Chen et al., 2022)	Faster R-CNN	VGG16	REPORTED	-	-	60.2	-	-	-	-	-	-
SFOD	MT (Tarvainen & Valpola, 2017)	Faster R-CNN	ResNet-50	REPORTED	-	-	41.2	-	-	-	-	-	-
	SED (Li et al., 2021)	Faster R-CNN	VGG16	REPORTED	-	-	43.6	-	-	-	-	-	-
	SED-Mosaic (Li et al., 2021)	Faster R-CNN	VGG16	REPORTED	-	-	44.6	-	-	-	-	-	-
	SOAP (Xiong et al., 2022)	Faster R-CNN	VGG16	REPORTED	-	-	42.7	-	-	-	-	-	-
	LODS (Li et al., 2022a)	Faster R-CNN	VGG16	REPORTED	-	-	43.9	-	-	-	-	-	-
	A^{2}SFOD (Chu et al., 2023)	Faster R-CNN	VGG16	REPORTED	-	-	44.9	-	-	-	-	-	-
	IRG (VS et al., 2023)	Faster R-CNN	ResNet-50	REPORTED	-	-	46.9	-	-	-	-	-	-
	RPL (Zhang et al., 2023)	Faster R-CNN	VGG16	REPORTED	-	-	47.8	-	-	-	-	-	-
	PETS (Liu et al., 2023)	Faster R-CNN	VGG16	REPORTED	-	-	47.0	-	-	-	-	-	-
	LPU (Chen et al., 2023)	Faster R-CNN	VGG16	REPORTED	-	-	48.4	-	-	-	-	-	-
	LPLD (Yoon et al., 2024)	Faster R-CNN	ResNet-50	REPORTED	-	-	51.3	-	-	-	-	-	-
	AdaBN (Li et al., 2018)	Faster R-CNN	VGG16-BN	REPORTED	-	-	37.5	-	-	-	-	-	-
	SF-UT (Hao et al., 2024)	Faster R-CNN	VGG16-BN	REPORTED	-	-	46.2	-	-	-	-	-	-
	SF-PL (Hao et al., 2024)	Mask R-CNN	ResNet-50	REPORTED	0.0	0.0	0.1	0.0	1.0	0.0	0.0	0.1	0.1
	SF-FM (Hao et al., 2024)	Mask R-CNN	ResNet-50	REPORTED	0.0	0.0	0.0	0.7	0.0	0.0	0.0	0.1	0.1
	SF-FM† (Hao et al., 2024)	Mask R-CNN	ResNet-50	REPORTED	0.0	0.0	8.6	0.0	0.0	0.0	0.0	1.1	1.1
	SF-UT (Hao et al., 2024)	Mask R-CNN	ResNet-50	REPORTED	20.2	33.3	52.9	12.0	**4.3**	**5.0**	3.2	23.9	19.3
	AdaBN + Fixed SF-PL (Hao et al., 2024)	Mask R-CNN	ResNet-50	REPORTED	28.8	34.9	62.7	16.6	2.0	1.2	7.3	31.5	23.1
	AdaBN + Fixed SF-PL † (Hao et al., 2024)	Mask R-CNN	ResNet-50	REPORTED	31.7	36.1	64.4	15.8	0.0	**5.0**	10.0	35.4	24.8
	AdaBN + Fixed SF-FM (Hao et al., 2024)	Mask R-CNN	ResNet-50	REPORTED	26.2	33.0	62.2	16.9	0.0	0.5	**12.4**	31.2	22.8
	AdaBN + Fixed SF-FM † (Hao et al., 2024)	Mask R-CNN	ResNet-50	REPORTED	35.8	37.3	63.3	16.2	0.0	2.5	6.6	35.1	24.6
	RPM †	Mask R-CNN	ResNet-50	BEST	**37.2**	**43.0**	64.4	**19.7**	0.0	**5.0**	3.8	**38.5**	**26.4**
	RPM + Uniform†	Mask R-CNN	ResNet-50	BEST	35.9	40.5	**64.8**	17.2	0.0	0.0	7.4	36.6	25.3
	RPM + Decoupled†	Mask R-CNN	ResNet-50	BEST	33.1	38.0	63.5	18.0	0.0	0.0	8.0	37.4	24.7
	SF-PL (Hao et al., 2024)	Mask R-CNN	ResNet-50	LAST	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0	0.0
	SF-FM (Hao et al., 2024)	Mask R-CNN	ResNet-50	LAST	0.0	0.0	0.0	0.1	0.0	0.0	0.0	0.0	0.0
	SF-FM† (Hao et al., 2024)	Mask R-CNN	ResNet-50	LAST	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	SF-UT (Hao et al., 2024)	Mask R-CNN	ResNet-50	LAST	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
	AdaBN + Fixed SF-PL (Hao et al., 2024)	Mask R-CNN	ResNet-50	LAST	23.7	26.8	57.7	**16.6**	0.0	2.5	5.0	26.8	19.9
	AdaBN + Fixed SF-PL † (Hao et al., 2024)	Mask R-CNN	ResNet-50	LAST	25.3	27.5	59.5	14.8	0.0	2.5	**8.3**	30.3	21.0
	AdaBN + Fixed SF-FM (Hao et al., 2024)	Mask R-CNN	ResNet-50	LAST	27.0	28.8	59.8	13.7	0.0	1.7	3.8	29.1	20.5
	AdaBN + Fixed SF-FM † (Hao et al., 2024)	Mask R-CNN	ResNet-50	LAST	28.9	28.8	62.0	15.4	0.0	2.5	5.1	30.5	21.6
	RPM †	Mask R-CNN	ResNet-50	LAST	**34.4**	**37.9**	**64.3**	13.9	0.0	**5.0**	6.3	36.0	**24.7**
	RPM + Uniform†	Mask R-CNN	ResNet-50	LAST	34.2	37.7	63.4	16.5	0.0	0.0	7.5	**36.6**	24.5
	RPM + Decoupled†	Mask R-CNN	ResNet-50	LAST	34.3	37.6	63.5	13.9	0.0	0.0	7.0	**36.6**	24.1
Oracle	Oracle (Hao et al., 2024)	Faster R-CNN	VGG16	—	-	-	49.9	-	-	-	-	-	-
	Oracle (Yoon et al., 2024)	Faster R-CNN	ResNet-50	—	-	-	58.5	-	-	-	-	-	-

Methods marked † use mask-refined pseudo-labels.