CD-Buffer: Complementary Dual-Buffer Framework for Test-Time Adaptation in Adverse Weather Object Detection

Supplementary Material

A. Overview

This supplementary material offers comprehensive details and additional experimental evidence supporting our main contributions. Section B describes implementation specifics, covering source model training, baseline method configurations, and additional CD-Buffer details. Section C analyzes CD-Buffer's design choices and introduces CD-Buffer Light, an efficient variant optimized for real-time deployment, along with its performance evaluation across various adaptation scenarios. Section D presents supplementary experimental results, including visualizations of the continual test-time adaptation process and qualitative detection results under diverse weather conditions.

B. Additional Implementation Details

Hardware: All experiments, including source model training and test-time adaptation, were performed using a single NVIDIA RTX 6000 Ada Generation GPU.

Source model training: For TTA experiments, we trained source model on the KITTI and Cityscapes datasets using the following configurations. All source models adopt the Faster R-CNN detector with a ResNet-50 [2] backbone initialized from ImageNet-pretrained weights [1]. We trained the models using the Adam optimizer with a batch size of 32 for a total of 250 epochs. The learning rates were set to 1e-4 for KITTI and 5e-4 for Cityscapes, respectively. The datasets contain eight object categories each:

- **KITTI:** Car, Van, Truck, Pedestrian, Person sitting, Cyclist, Tram, Misc.
- **Cityscapes:** Person, Rider, Car, Truck, Bus, Train, Motorcycle, Bicycle.

The resulting source model performance, measured by mAP@50, is 85.75 on KITTI, and 24.75 on Cityscapes.

Baselines implementation: We provide detailed implementation settings for all baseline methods. For fair comparison, we unified the batch size to 16 across all experiments and selected learning rates by testing both the originally reported values and the learning rate used in our method, choosing whichever yielded better performance.

BufferTTA. Following the original paper's best-performing configuration, we introduce parallel 1×1 and 3×3 convolutional layers as buffer layers. These buffer layers are inserted after ReLU activation functions. Among four stages of ResNet-50 backbone, we experimented with buffer placement in each stage and selected Stage 3, which achieved the highest performance. The learning rate is set to 1×10^{-5} uniformly across all datasets. Learnable parameters include batch normalization affine parameters and the introduced buffer parameters.

Algorithm 1 CD-Buffer Test-Time Adaptation

Require: Source domain statistics $\{\bar{X}_s^l, \bar{x}_s^l, \bar{\mu}_s^l, \bar{\sigma}_s^l\}$, $\lambda_S, \lambda_A, \lambda_{\text{reg}}, \rho_{\text{target}}, r$.

Initialize mask scores $s = |\gamma|$.
for each target batch $\mathcal{B}_t$ in the test stream **do**
  for each adaptable block **do**
    Let F_{in} denote the input feature of the adaptable block.
    for each layer l **do**
      Let X_t^l denote the input feature of layer l.
      Compute m_{hard} and m for the subtractive buffer as in Eqs. (6)–(8).
      Step1: Subtractive buffer:
      Compute F_{out}^s as in Eq. (4).
    end for
    Compute $\hat{m}_{\text{soft}}^{-1}$ for the additive buffer as in Eqs. (10) and (11).
    Step2: Additive buffer:
    Compute $F_{\text{out}}^{\text{a}}$ as in Eqs. (9) and (12).
    $F_{\text{out}} = F_{\text{out}}^s + F_{\text{out}}^{\text{a}}$.
  end for
  Step3: Compute losses
  Compute domain discrepancy score D as in Eqs. (1)–(3).
  Compute total loss $\mathcal{L}_{\text{total}}$ as in Eqs. (5), (13), and (14).
  Step4: Optimization
  Compute D_l as in Eq. (15).
  Update mask scores s^l and BN layer parameters.
  Update additive buffer parameters scaled by D_l.
  Step5: Stochastic reactivation
  for each suppressed channel with $m^{c,l} = 0$ **do**
    Sample a random variable v_c^l with
    $v_c^l \sim \text{Bernoulli}(r)$.
    if $v_c^l = 1$ **then**
      $s_c^l = |\gamma_c^l|$.
    end if
  end for
end for

Pruning TTA. Since official code is not publicly available, we implement this method following the paper's description. The approach is applied to all BN layers in the ResNet-50 backbone. For fair comparison, we set the pruning ra-

tio threshold to 0.05, matching our suppression ratio. The learning rate is set to 1×10^{-4}. All other hyperparameters follow the reported values in the paper. Learnable parameters consist of scaling factors in BN layers while freezing all other parameters. The loss function employs KL divergence between source and target feature distributions for all adaptable BN layer inputs, using both image-level and instance-level features.

ActMAD. Apply L1 loss between source and target batch statistics (mean and variance) for all BN layer output features. Unlike other methods, all model parameters are updated during adaptation. The learning rate is set to 1×10^{-4}. All other hyperparameter settings follow the configurations reported in the original paper.

WHW. This method aligns source and target feature distributions using KL divergence loss, utilizing both image-level and instance-level features. For stable statistical estimation of target feature statistics, we employ exponential moving average (EMA). The learning rate is set to 1×10^{-4} for all datasets. All other hyperparameters follow the reported settings.

Additional CD-Buffer Details: CD-Buffer is applied to all basic blocks across all stages of the ResNet-50 backbone network. Within each block, the subtractive buffer is applied to every BN layer, while one additive buffer is applied per block. The scaling factor k for determining the range of $\hat{m}_{\text{soft}}^{-1}$ is set to 10, resulting in adaptive channel-wise scaling within the range $[0, 10]$ for the additive buffer. To prevent early suppression, we introduce a stochastic reactivation mechanism following the approach in prior work [3]. At each adaptation step, suppressed channels (where $m_c = 0$) are reactivated with probability r through Bernoulli sampling:

$$v_c \sim \text{Bernoulli}(r), \quad \text{if } v_c = 1 : s_c \leftarrow |\gamma_c|, \tag{A.1}$$

where v_c is a random variable determining reactivation, and the mask score s_c is reset to the absolute value of the corresponding BN parameter γ_c. The reactivation probability is set to $r = 0.05$ across all experiments. The complete TTA process is presented in Algorithm 1.

C. CD-buffer Light: Efficient Variant for Real-Time Applications

Given the real-time nature of object detection tasks, processing speed (FPS) is as critical as adaptation performance. While applying CD-Buffer to all stages of the backbone network achieves optimal performance, it may introduce computational overhead unsuitable for time-critical applications. Therefore, we conduct an ablation study on CD-Buffer placement across different stages and propose CD-Buffer Light, a more efficient variant tailored for real-world scenarios.

C.1. CD-buffer Placement Analysis

Table 1. presents TTA results on KITTI and Cityscapes datasets, comparing our fully adaptation model (applied to all stages) against variants where CD-Buffer is applied to only one stage (Stage 1, 2, 3, or 4) of the ResNet-50 backbone. Single-stage variants exhibit performance degradation compared to the fully adaptation model, but most configurations still outperform baseline methods. Notably, Stage 4 shows the weakest performance among single-stage variants. In contrast, Stage 1 demonstrates the most consistent performance across all severity levels, achieving the best average performance among single-stage variants. Moreover, comparing with baseline methods, Stage 1 outperforms most baselines across nearly all conditions. This demonstrates that our framework's performance does not degrade linearly with the number of learnable parameters, maintaining competitive results even with significantly reduced parameters. Based on these observations, we propose **CD-Buffer Light**, which applies our framework exclusively to Stage 1, as it provides the optimal balance between adaptation effectiveness and computational efficiency.

C.2. Performance Evaluation of CD-Buffer Light

Tables 2 and 3 present additional experimental results for CD-Buffer Light on continual adaptation scenarios and ACDC, respectively. Across both settings, CD-Buffer Light consistently outperforms baseline methods, validating its effectiveness despite reduced computational cost. While performance is slightly lower than the full model, the gap is modest.

C.3. Inference Speed Comparison

Table 4 compares inference speed in FPS for all baseline methods, our fully adaptation model, and Ours-Light. The full CD-Buffer model is slower than most baselines due to buffers applied across all basic blocks. In contrast, CD-Buffer Light achieves significantly faster inference, outperforming all methods except Pruning TTA while maintaining superior adaptation performance.

Our framework offers flexibility for different deployment scenarios. CD-Buffer (Full) is recommended when accuracy is prioritized, while CD-Buffer Light suits real-time applications with strict latency constraints.

D. Additional Results

Continual TTA visualization on KITTI dataset

Figure 1 visualizes the adaptation dynamics under cyclic domain shifts, where fog severity repeatedly transitions between 50m, 75m, and 150m over 10 rounds on the KITTI dataset. Our method maintains consistently superior performance throughout all cycles, demonstrating stable and

	KITTI fog			KITTI rain			Cityscapes foggy	
corruption/severity	**fog/50m**	**fog/75m**	**fog/150m**	**rain/200mm**	**rain/100mm**	**rain/75mm**	**fog/0.02**	**fog/0.01**
Ours	**44.80**	**56.06**	**68.42**	**63.22**	71.40	73.03	**18.39**	**21.79**
Stage 4	36.91	44.69	58.79	53.65	65.11	67.19	12.50	17.25
Stage 3	**42.19**	**50.89**	**65.64**	**62.21**	70.70	72.05	**17.48**	**22.51**
Stage 2	**43.28**	**53.99**	**67.26**	**63.09**	71.39	72.58	**17.94**	20.63
Stage 1(Ours-Light)	**42.51**	**52.04**	**66.95**	**61.85**	70.88	73.08	**18.57**	**22.49**

Table 1. Quantitative results of object detection under various adverse weather conditions. Performance (mAP@50) is reported for KITTI → KITTI (fog, rain) and Cityscapes → Cityscapes Foggy scenarios with different corruption severities. We highlight entries that still achieve the best performance compared to the baselines. in bold, and those with the second-best performance with an underline. We propose Stage 1 as Ours-Light, which consistently maintains strong performance across various corruptions and severity levels.

	KITTI → KITTI fog (50m → 75m → 150m)												
	Round1			Round2			Round5			Round10			Average
	50m	75m	150m	50m	75m	150m	50m	75m	150m	50m	75m	150m	
Ours	34.90	52.70	**69.93**	**44.51**	**57.29**	**70.72**	**46.53**	**58.30**	68.02	**45.40**	**56.14**	**63.51**	**55.66**
Ours-Light	30.18	49.10	**67.92**	40.36	**53.52**	**69.14**	**45.52**	**58.02**	**70.24**	**47.20**	**59.54**	**68.49**	**54.94**

Table 2. Continual test-time adaptation (TTA) results for the KITTI → KITTI fog scenario. This experiment simulates progressive domain shift where fog severity gradually increases (50 m → 75 m → 150 m). Each model continuously adapts over 10 rounds, using outputs from one severity level as initialization for the next. We highlight entries that still achieve the best performance compared to the baselines. in bold, and those with the second-best performance with an underline.

	Cityscapes → ACDC			
corruption	**fog**	**snow**	**rain**	**night**
Ours	**24.45**	15.41	**13.71**	**8.92**
Ours-Light	24.15	16.97	**15.15**	**8.18**

Table 3. Object detection results (mAP@50) for **Cityscapes → ACDC** across four corruption types (fog, snow, rain, night). We highlight entries that still achieve the best performance compared to the baselines in Table. in bold, and those with the second-best performance with an underline.

	FPS
Direct Test	73.24
Buffer TTA	33.69
Pruning TTA	**40.62**
ActMAD	35.59
WHW	31.38
Ours	15.59
Ours_Light	35.86

Table 4. Inference speed comparison (FPS) for different TTA methods measured on a dummy input. The highest FPS is highlighted in bold, and the second-highest is underlined.

robust adaptation across repeated domain transitions.

Qualitative results Figures 3, 4, and 5 present qualitative visualization of object detection results across diverse adverse weather conditions. We compare three settings: Direct Test (source model without adaptation), Ground Truth annotations, and our CD-Buffer adaptation results. The visualizations cover KITTI fog and rain scenarios at various severity levels, Cityscapes fog conditions, and ACDC including fog, snow, rain, and night conditions. The qualitative results clearly demonstrate that our method effectively addresses domain shift challenges at test time.

Fog	Buffer TTA (Add)	Pruning TTA (Sub)	Parallel	Ours
30m (Severe)	14.54	23.89	18.58 (**+4.04**/ **-5.31**)	**33.04** (**+18.50**/ **+9.15**)
375m (Mild)	73.70	70.60	72.11 (**-1.59**/ **+1.51**)	**74.97** (**+1.27**/ **+4.37**)

Table 5. Parallel combination vs CD-Buffer on KITTI.

Analysis of Complementary Behaviors.

Our contribution lies not in the additive and subtractive components themselves, but in how they are coupled through a unified discrepancy metric. Prior methods treat them as independent alternatives, but we reformulate this as a channel-wise control problem. In Tab. 5, additive and subtractive methods exhibit opposite performance trends depending on severity, and their parallel combination fails to resolve this trade-off and can even degrade performance in certain cases (**red**). CD-Buffer achieves the best performance under both severe and mild conditions. The inverse soft mask enables automatic balancing, where suppression extent directly modulates compensation strength. In practical TTA where method selection cannot be made at runtime, consistent performance across diverse severities is a meaningful contribution.

Complementarity is further characterized by severity-

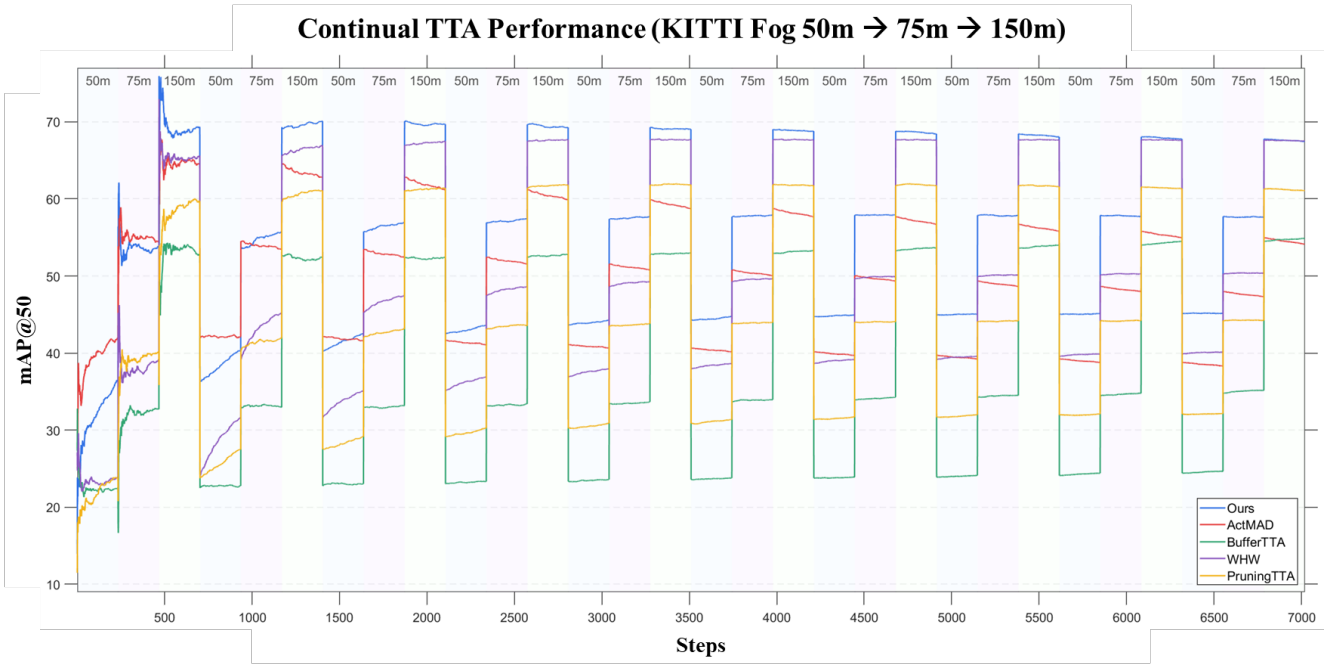

Figure 1. **Visualization of continual TTA process on KITTI fog with cyclic severity transitions** (50m → 75m → 150m, repeated 10 times). Our method maintains consistently superior performance across all cycles, demonstrating both rapid adaptation within each severity level and robustness to various severity.

	30m	50m	75m	150m
w/o Sub	**31.88(-1.18)**	**44.51(-1.15)**	55.67(-0.81)	67.88(-0.45)
w/o Add	32.96(-0.10)	45.22(-0.44)	**55.18(-1.30)**	**65.75(-2.58)**
Full	33.06	45.66	56.48	68.33

Table 6. Severity-wise ablation on KITTI fog.

dependent roles rather than uniform contribution. In Tab. 6, the subtractive module contributes more under severe shifts (30m, 50m), while the additive module is more effective under milder shifts (75m, 150m), and the full model achieves the most consistent performance across conditions. The modest gain (+0.08) observed in main paper can be attributed to the use of Cityscapes-Fog, which predominantly exhibits mild shifts. Under such conditions, the additive module naturally dominates, consistent with our analysis. The primary benefit of the proposed approach lies in stable adaptation across diverse shift severities rather than large average gains.

	L1	L2	Cosine
mAP50	**63.61**	63.17	63.50

Table 7. Comparison of discrepancy metrics.

Validation of the Discrepancy Metric.

To justify the choice of the discrepancy metric, we adopt L1 as a simple and effective measure of shift magnitude,

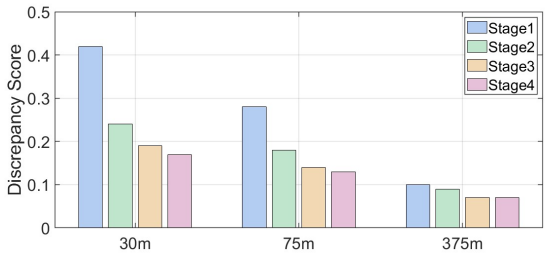

Figure 2. Discrepancy across fog severities and backbone stages.

as supported by prior TTA methods (e.g., ActMAD, PruningTTA). In Fig. 2, the discrepancy consistently decreases as visibility increases, confirming that the metric reliably reflects domain shift. A stage-wise analysis further shows that Stage 1 exhibits the highest discrepancy; applying CD-Buffer only to Stage 1 achieves the best single-stage performance Tab. 1, indicating that the metric also captures location-specific shifts. In Tab. 7, we compare L1 with L2 and cosine similarity. While the differences are marginal, L1 achieves the best performance, supporting it as a simple yet effective and empirically validated choice for our framework.

Scope and Generalization.

We evaluate the proposed method on a Swin-T Transformer-based detector, where BN is replaced with

	Direct Test	ActMAD	WHW	Ours
200mm	15.30	24.81	24.90	**28.05**
100mm	27.37	35.55	32.65	**36.93**
75mm	31.88	38.61	38.71	**40.06**

Table 8. Transformer-based detector (Swin-T) on KITTI rain.

	Direct Test	Buffer TTA	Pruning TTA	ActMAD	WHW	Ours
Gaussian	8.87	11.69	31.60	40.14	23.15	**43.43**
Jpeg	65.66	70.65	66.02	62.65	**72.45**	70.59
Motion	47.81	51.55	63.49	63.74	64.49	**72.37**
Defocus	52.6	57.14	63.27	61.81	66.77	**70.7**

Table 9. Evaluation on non-weather corruptions(KITTI).

LN and MLP-based adapters are used (Tab. 8). The results show consistent improvements over baseline methods across all shift severities. The ACDC benchmark already includes compound domain shifts, such as sensor differences between datasets and diverse appearance variations (e.g., brightness, blur, and contrast). To further assess generalization, we additionally evaluate on non-weather corruptions, including Gaussian noise, JPEG compression, motion blur, and defocus blur (Tab. 9), where our method achieves the best or comparable performance.

	Direct Test	Buffer TTA	Pruning TTA	ActMAD	WHW	Ours
30m	12.37	14.54	23.89	28.31	14.79	**33.04**
375m	69.57	73.70	70.60	65.55	72.73	**74.97**
750m	74.67	78.09	74.89	68.07	77.65	**78.58**

Table 10. Extended severity evaluation on KITTI fog.

Performance Across Extended Visibility Ranges.

We extend the evaluation to a broader range of visibility conditions beyond the primary range (Tab. 10). Under milder conditions ($\geq$375m), the direct test performance is already close to the clean setting, leaving limited room for improvement. The results show that subtractive approaches contribute more under severe shifts (30m), while additive approaches are more effective under milder shifts (375m, 750m), consistent with the complementary pattern observed in Fig. 1 (main paper). Our method achieves the best performance across all visibility levels.

References

[1] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*, pages 248–255. Ieee, 2009. 1

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[3] Kunyu Wang, Xueyang Fu, Xin Lu, Chengjie Ge, Chengzhi Cao, Wei Zhai, and Zheng-Jun Zha. Efficient test-time adaptive object detection via sensitivity-guided pruning. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pages 10577–10586, 2025. 2

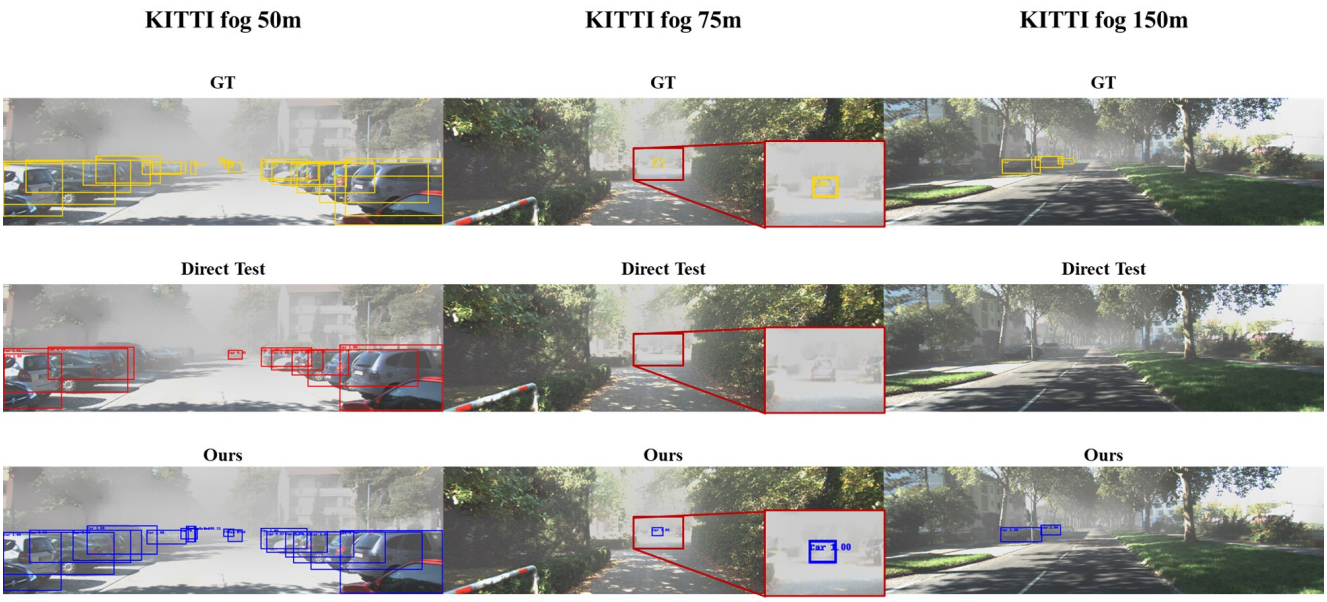

Figure 3. **Qualitative results on KITTI fog.** Object detection results visualized with bounding boxes across different fog severities (50m, 75m, 150m). We compare ground truth (GT), Direct Test (source model without adaptation), and our CD-Buffer results, demonstrating effective adaptation across all severity levels.

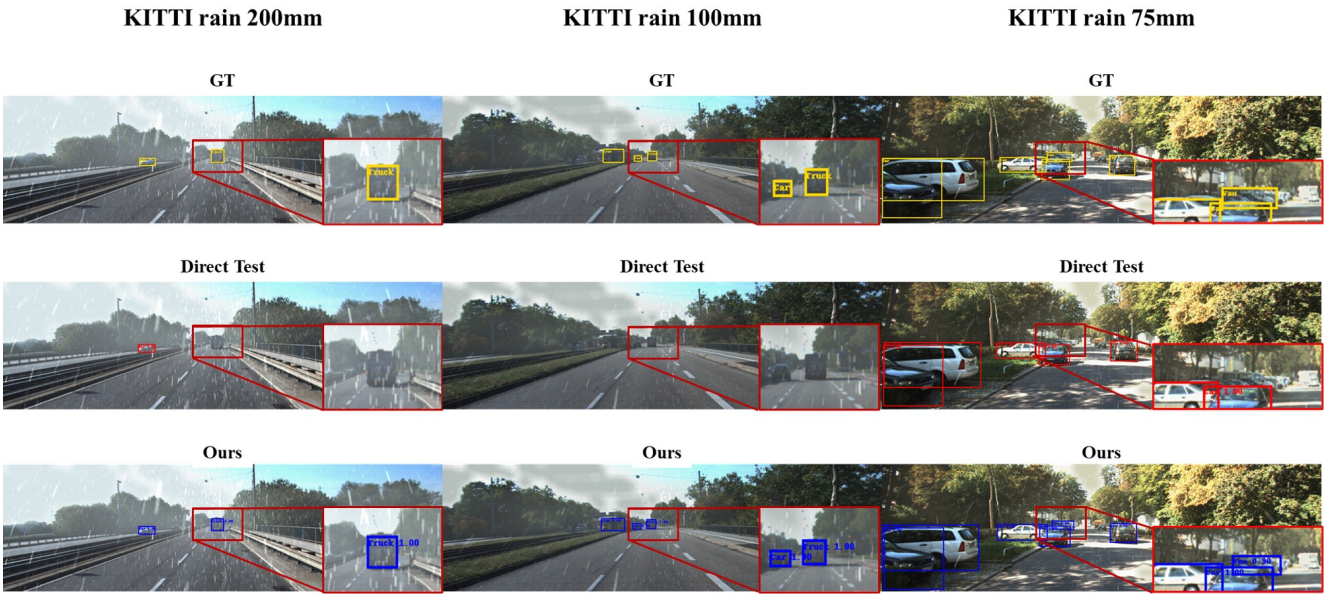

Figure 4. **Qualitative results on KITTI rain.** Object detection results visualized with bounding boxes across different rain severities (200mm, 100mm, 75mm). We compare ground truth (GT), Direct Test (source model without adaptation), and our CD-Buffer results, demonstrating effective adaptation across all severity levels.

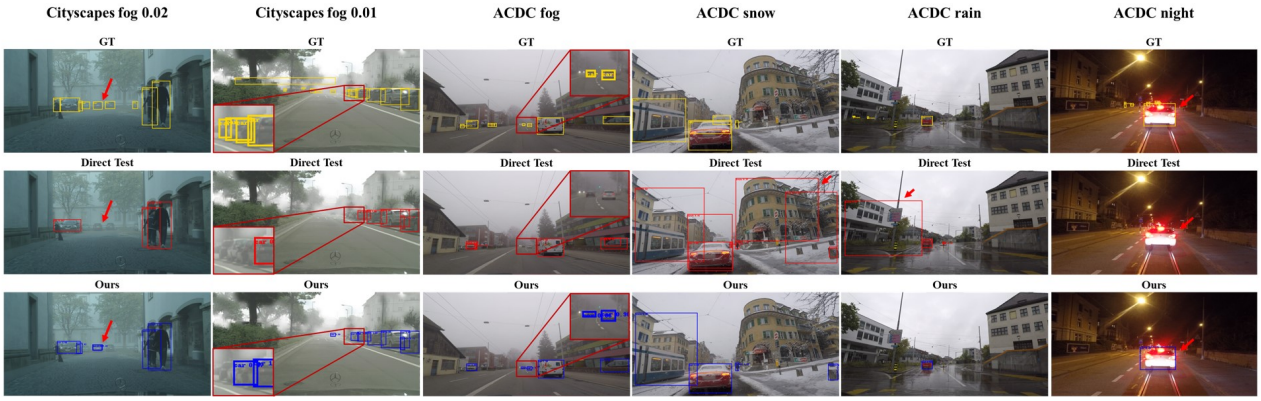

Figure 5. **Qualitative results on Cityscapes fog and ACDC.** Object detection results visualized with bounding boxes. For Cityscapes, we show fog severities (0.02, 0.01), and for ACDC, we present diverse conditions including fog, snow, rain, and night. We compare ground truth (GT), Direct Test (source model without adaptation), and our CD-Buffer results, demonstrating effective adaptation across all conditions.