CROTTA: CROSS-TASK AND CROSS-FRAME TEST-TIME ADAPTATION FOR END-TO-END DRIVING

Anonymous authors
Paper under double-blind review

ABSTRACT

End-to-end autonomous driving has achieved significant milestones, yet its reliance on the i.i.d. assumption renders it vulnerable to real-world distribution shifts. While Test-Time Adaptation (TTA) offers a promising paradigm for on-the-fly enhancement, its application to driving is hindered by two barriers: (i) *multi-task adaptation dependency*, where adapting tasks in isolation propagates uncoordinated updates, triggering task-level interference; and (ii) *continuous adaptation instability*, where recursive online updates over long horizons cause compounding parameter drift and model collapse. To address these challenges, we propose CroTTA, a plug-and-play TTA framework for the robust online adaptation of end-to-end driving models. CroTTA introduces a joint adaptation objective that integrates perception multi-modal uncertainty with planning temporal consistency; by formulating cross-task constraints, it ensures that adaptation signals from the upstream perception remain consistent with downstream planning, thereby enhancing the adaptation performance under diverse distribution shifts. Furthermore, to ensure continuous adaptation stability, we develop a teacher-student architecture where the teacher model guides student updates across two time scales: it maintains a global anchor via EMA to suppress long-term cumulative drift, while acting as a short-term gatekeeper to discard transient, destabilizing updates through a per-step filter. Extensive experiments across multiple models demonstrate that, by updating only $0.15‰$ of parameters with a 20ms latency, CroTTA achieves average recovery rates of 24.7%, 26.9%, and 11.8% for L2, collision rate, and PDMS, respectively, under severe distribution shifts. In comparison with existing TTA methods, CroTTA obtains an additional 20.5 and 18.2 percentage points of L2 and collision recovery. Our code is available at: `https://anonymous.4open.science/r/CroTTA-88D0/`.

1 INTRODUCTION

End-to-end autonomous driving has emerged as a prevalent paradigm, demonstrating remarkable proficiency and generalization potential across diverse driving scenarios (Chen et al., 2024b). However, the performance of such models relies heavily on the assumption that test and training data are independently and identically distributed (i.i.d.), which rarely holds in practice (Do et al., 2024; Shoeb et al., 2025). As shown in Figure 1(a), these models degrade sharply under real-world distribution shifts; for instance, dense fog, brightness fluctuations, or Gaussian noise can cause error rates to surge under severe corruption. Existing approaches address this through domain adaptation (DA) (Ben-David et al., 2006; Ning et al., 2025) or data augmentation (Zhong et al., 2020; Wang et al., 2021), yet these paradigms rely on target-domain annotations or costly retraining. Such requirements are often infeasible for real-time driving, where distribution shifts manifest as continuous, unpredictable streams that demand instantaneous adaptation (Malinin et al., 2021; Diehl et al., 2025). This motivates equipping autonomous driving models with the ability to adapt online at test time, requiring neither labeled data nor offline retraining.

Test-time adaptation (TTA) offers a compelling alternative by updating models on the fly using only unlabeled test samples (Wang et al., 2020; Liang et al., 2025). Early fully TTA (FTTA) methods adapt the model via lightweight self-supervised objectives (Shin & Kim, 2024; Xiao & Snoek, 2024; Karmanov et al., 2024). Continual TTA (ConTTA) methods further extend this by continuing to update model parameters across successive batches of test samples, introducing mechanisms to prevent forgetting (Park et al., 2025; Hoang et al., 2024; Tan et al., 2025). However, existing TTA

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

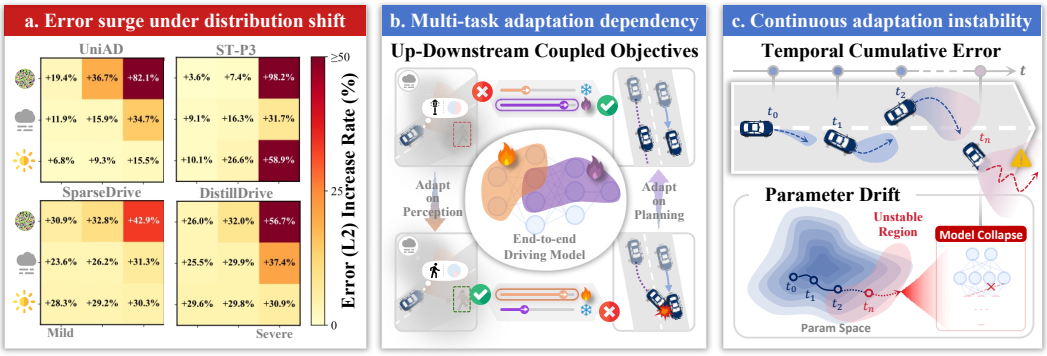


Figure 1: **Motivation and core challenges.** Distribution shifts cause severe performance degradation in end-to-end autonomous driving models, where effective test-time adaptation is further challenged by multi-task adaptation dependencies and continuous adaptation instability.

paradigms are predominantly designed for closed-set, single-task settings. Consequently, directly applying these methods to end-to-end autonomous driving under severe and continuous distribution shift remains largely unexplored.

Adapting end-to-end driving models at test time poses two fundamental challenges for existing TTA methods. (i) *Multi-task adaptation dependency.* End-to-end models couple objectives from perception to planning; adapting to one task in isolation can propagate uncoordinated updates to others, triggering potential gradient interference that undermines overall performance (Yu et al., 2020; Chen et al., 2018). (ii) *Continuous adaptation instability.* Autonomous driving involves long-horizon sequences where adaptation is a recursive process. However, continuous online adaptation introduces the compounding risk of parameter drift and model collapse, which becomes increasingly severe as updates accumulate.

To address these challenges, we propose Cross-Task and Cross-Frame Test-Time Adaptation (**CroTTA**) that allows end-to-end models to continuously update parameters during inference. To mitigate multi-task adaptation dependency, CroTTA constructs a joint adaptation objective over the upstream perception and downstream planning, coupling perception multi-modal uncertainty with planning temporal consistency. By formulating cross-task constraints, it transforms independent task signals into a unified adaptation force, allowing the tasks to provide synergetic guidance during adaptation. To mitigate continuous adaptation instability, CroTTA introduces a teacher-student architecture where the teacher regulates the student at two scales: it maintains a stable parameter anchor via EMA to constrain the student's update space, suppressing long-term cumulative drift. Simultaneously, it acts as a short-term gatekeeper that performs per-step degradation verification, discarding any student updates that increase planning loss. By leveraging this two-scale guidance, the architecture breaks the recursive error propagation and ensures reliable adaptation without collapse. Extensive experiments on nuScenes and NAVSIM under severe Gaussian noise corruption demonstrate that CroTTA generalizes across various end-to-end models of diverse architectures, surpassing existing TTA baselines by a significant margin. Moreover, ablation studies further validate the contribution of each component, and efficiency analysis confirms real-time deployability. In summary, our contributions are as follows:

- We propose CroTTA, a plug-and-play framework for end-to-end driving that enables diverse pretrained models to adapt at test time under severe and continuous distribution shifts.
- We design a joint adaptation objective that couples perception multi-modal uncertainty with planning temporal consistency, enabling cross-task coordination between upstream and downstream.
- We introduce a teacher-student architecture that combines long-term parameter anchoring with per-step update verification to suppress cumulative drift and prevent model collapse.
- We demonstrate that CroTTA updates only 0.15‰ of parameters with an average latency of 20ms, achieving average recovery rates of 24.7%, 26.9%, and 11.8% for L2, collision rate, and PDMS, respectively, under distribution shift.

2 Related Work

Domain Adaptation. DA addresses distribution shift by aligning features between the labeled source domain and the unlabeled target domain (Farahani et al., 2021; Sohn et al., 2017; You et al., 2019; Kodirov et al., 2015), typically through moment matching (Peng et al., 2019), adversarial domain discrimination (Tang & Jia, 2020; He et al., 2025), or optimal transport (Ganin et al., 2016; Courty et al., 2016), achieving strong performance when target data is available at training time. However, DA methods require access to target domain data during training and often necessitate retraining—making them impractical when the test distribution is unknown in advance.

Test-Time Training. Test-Time Training (TTT) introduces an auxiliary self-supervised task optimized using only the test sample, removing the need for target domain data (Sun et al., 2020; Dalal et al., 2025). By optimizing auxiliary objectives such as rotation prediction (Deng et al., 2021) or contrastive learning directly on the test input (Chen et al., 2022; Osowiechi et al., 2024; Xiao et al., 2020), TTT enables the model to update without any labeled data, improving robustness to unseen distributions. Nevertheless, the auxiliary task must be co-designed and co-trained with the main objective, requiring modifications to the training pipeline and making TTT incompatible with off-the-shelf pretrained models.

Fully Test-Time Adaptation. FTTA requires no modification to training, updating model parameters using unsupervised signals at test time (Wang et al., 2020; Zhao et al., 2023; Rafi et al., 2026; Chu et al., 2025). Entropy minimization over model predictions has emerged as the dominant paradigm, with subsequent work introducing various mechanisms to improve update reliability and prevent parameter drift (Tan et al., 2025; Zhang et al., 2022; Hoang et al., 2024). Despite their practicality, existing FTTA methods operate on each test sample independently, without accounting for the need to sustain reliable adaptation over a long sequence (Niu et al., 2022; Song et al., 2023).

Continual Test-Time Adaptation. ConTTA extends FTTA to settings where the model must update over a sequence of test samples, requiring adaptation without forgetting (Niu et al., 2023; Wang et al., 2022). To prevent error accumulation, recent methods maintain an exponential moving average (EMA) model to generate stable pseudo-targets and stochastically restore a subset of parameters toward their pretrained values to preserve source knowledge (Sójka et al., 2023; Ye et al., 2025; Tan et al., 2026; Chen et al., 2024a). Yet despite the name, *continual* in existing ConTTA methods typically refers to the accumulation of updates across discrete samples, failing to capture the intrinsic time dependencies in driving sequences. Furthermore, these methods do not account for the joint adaptation of multi-task objectives in end-to-end models. Consequently, the unique combination of continuous evolution and multi-task coupling in autonomous driving presents challenges that existing TTA paradigms are not designed to address.

3 Problem Definition

Table 1: **Comparison of adaptation paradigms under distribution shift.** Existing methods fail to jointly address cross-task and cross-frame dependencies inherent to end-to-end driving models. Our method bridges this gap, enabling reliable test-time adaptation under continuous distribution shift. Blue-shaded row denotes our method. x: unlabeled test sample; $\mathcal{L}_{sup}$/$\mathcal{L}_{align}$/$\mathcal{L}_{aux}$/$\mathcal{L}_{self}$: supervised loss / domain alignment loss / auxiliary self-supervised loss / self-supervised objective.

Method	Adaptation Phase	Objective	Source Acc.	Cross-task	Cross-frame
DA	Train	$\mathcal{L}_{sup} + \mathcal{L}_{align}$	✓	-	-
TTT	Test + Train	$\mathcal{L}_{sup} + \mathcal{L}_{aux}(x^t)$	-	-	-
FTTA	Test	$\mathcal{L}_{self}(x^t)$	-	-	-
ConTTA	Test	$\mathcal{L}_{self}(x^t)$	✓	-	-
CroTTA (Ours)	Test	$\mathcal{L}_{self}(x^t)$	✓	✓	✓

Table 1 summarizes existing adaptation paradigms and highlights the two properties that existing TTA methods leave unaddressed. Current TTA methods share a common update rule: given an unlabeled test sample x_s at step s, the affine normalization parameters in BatchNorm θ^{BN} are updated by minimizing a self-supervised objective $\mathcal{L}_{self}$,

$$\theta_s^{BN} = \theta_{s-1}^{BN} - \eta \nabla_\theta \mathcal{L}_{self}(x_k;\, \theta_{s-1}^{BN}), \tag{1}$$

where η is the learning rate. While existing TTA research treats the model as a direct mapping from input x_s to output, end-to-end driving models are structured as a cascade of upstream perception to downstream planning, where the planning output $\hat{\mathcal{Y}}_{plan}$ is not determined by raw input alone but is mediated through the intermediate perception feature $\mathcal{Z}_{perc}$ via a downstream mapping f_{prog}. Since autonomous driving processes inputs as a continuous sequence x_t indexed by time t, the adaptation naturally targets LayerNorm parameters θ^{LN}, and the upstream-to-downstream dependency takes the form:

$$\hat{\mathcal{Y}}_{plan} = f_{prog}\big(x_t \mid \mathcal{Z}_{perc}(x_t;\, \theta^{LN}),\, \theta^{LN}\big). \tag{2}$$

Under distribution shift, corruption in $\mathcal{Z}_{perc}$ propagates through f_{prog}, causing errors to cascade from perception to planning; conversely, shared parameters couple gradient signals back across tasks.

Furthermore, while continual TTA methods introduce multi-step updates to handle large-scale inputs, they typically treat each sample in frame isolation. Autonomous driving, by contrast, presents a continuously evolving sequence of inputs under persistent and severe distribution shift, where parameter updates accumulate over time and errors compound across the sequence if left unchecked (Codevilla et al., 2019; 2018; Zhao et al., 2026). Therefore, TTA for end-to-end driving must jointly address cross-task dependency across coupled upstream perception and downstream planning, and cross-frame dependency arising from continuous adaptation over an evolving sequence:

$$\theta_t^{LN} = \mathcal{P}_\Omega\Bigg(\theta_{t-1}^{LN} - \eta\nabla_\theta \mathcal{L}_{joint}\Big(x_t,\, \underbrace{\mathcal{Z}_{perc}(x_t)}_{\text{Upstream}},\, \underbrace{\hat{\mathcal{Y}}_{plan}}_{\text{Downstream}}\,;\, \theta_{t-1}^{LN}\Big)\Bigg). \tag{3}$$

Here $\mathcal{L}_{joint}$ coordinates adaptation across the upstream and downstream objectives, and $\mathcal{P}_\Omega$ denotes the projection operator onto the parameter constraint space Ω to ensure update stability across the continuous test sequence.

4 Methodology

To simultaneously resolve the cross-task dependency and cross-frame dependency formalized in Eq. 3, CroTTA introduces two complementary components. Taking into account the coupling relationship between upstream perception and downstream planning, CroTTA constructs a joint self-supervised objective $\mathcal{L}_{joint}$ to enable unified cross-task coordination throughout the adaptation process. Simultaneously, a teacher-student stability framework supervises the continuous adaptation process at two time levels to prevent error accumulation across frames, resolving cross-frame dependency. Both components update only the LayerNorm affine parameters without any other modification to the pretrained model, making CroTTA plug-and-play across diverse end-to-end driving architectures.

4.1 Joint Adaptation Objective for Cross-Task Synergy

End-to-end autonomous driving models are inherently multi-task systems with highly interdependent components (Grigorescu et al., 2020; Chitta et al., 2022; Huang et al., 2023). While supervised training aligns diverse task objectives toward a unified ground-truth, ensuring that improvements in one component benefit the entire pipeline, TTA lacks such a definitive reference to anchor the synergistic updates between coupled tasks. Consequently, optimizing a singular task-specific proxy can easily lead to blind parameter updates that disrupt the established mapping equilibrium, where small shifts are amplified through the cascaded architecture and manifest as severe cross-task interference. To bridge this gap, we propose a joint adaptation objective that regularizes the parameter updates through two complementary forces:

$$\mathcal{L}_{joint} = \mathcal{L}_{mm} + \lambda\mathcal{L}_{temp}, \tag{4}$$

where $\mathcal{L}_{mm}$ characterizes uncertainty derived from the multi-modal entropy on the perception side, and $\mathcal{L}_{temp}$ imposes a self-supervised constraint on the planning side by exploiting the observability of cross-frame displacement, with λ controlling its relative weight. By coupling these terms, we prevent blind parameter updates from disrupting the model's internal synergy, thereby maintaining a cross-task equilibrium.

Upstream Perception: Multi-modal Entropy Minimization. Under severe environmental perturbations—such as dense fog or sensor noise—the model's perceptual discriminability degrades,

causing the output probability distributions to undergo a flattening effect that manifests as heightened predictive uncertainty (Hendrycks & Dietterich, 2019; Ovadia et al., 2019). This flattening indicates that the latent embeddings have drifted into low-density regions of the source-domain manifold, making it difficult for the model to produce decisive categorical or attribute assignments.

To quantify the uncertainty arising from the distortion of $\mathcal{Z}_{perc}$, we formalize the output process of the perception heads. At each time step t, the system generates N parallel perceptual responses (e.g., object detection, semantic occupancy) based on the latent features, where the normalized confidence $q_{t,i}$ for the i-th response is mapped via:

$$q_{t,i} = \sigma\big(\Phi^{(i)}(\mathcal{Z}_{perc}(x_t, \mathcal{H}_{t-1}; \theta_{t-1}))\big) \tag{5}$$

Here, $\sigma(\cdot)$ denotes the sigmoid function, $\Phi^{(i)}$ denotes the decoder head for the i-th perceptual modality, while $\mathcal{H}_{t-1}$ represents the hidden state that propagates historical context. Since these perceptual modalities produce independent probabilistic outputs rather than a single mutually exclusive category, we define the binary information entropy to measure the uncertainty of each inference:

$$\mathbb{E}_{ent}(q_{t,i}) = -\Big(q_{t,i}\log(q_{t,i}) + (1-q_{t,i})\log(1-q_{t,i})\Big) \tag{6}$$

The multi-modal uncertainty loss $\mathcal{L}_{mm}$ is then derived from the expected entropy across all N parallel perceptual modalities:

$$\mathcal{L}_{mm} = \frac{1}{N}\sum_{i=1}^{N}\mathbb{E}_{ent}(q_{t,i}) \tag{7}$$

By minimizing $\mathcal{L}_{mm}$, the model is encouraged to recalibrate its latent representations toward high-density clusters, thereby restoring categorical decisiveness.

Downstream Planning: Temporal Consistency Optimization. Crucially, solely minimizing perceptual entropy may lead to *decisive but erroneous* interpretations, as the model might confidently converge toward incorrect scene semantics. To counter this, we exploit the inherent temporal continuity of ego-planning as a self-supervised signal. This approach is grounded in the physical reality that a vehicle's trajectory exhibits continuity, meaning the future states predicted at the previous timestamp should remain consistent with the actual ego-state observed at the current moment.

Therefore, to formulate the temporal prior, we utilize the planning trajectory generated at the previous timestamp. Let $\hat{p}_t$ denote the prior expectation of the ego-state for the current time t, which is the first-step prediction within the planning horizon $\hat{\mathcal{Y}}_{plan}^{(t-1)}$ generated at $t-1$:

$$\hat{p}_t = f_{plan}(x_{t-1}, \mathcal{H}_{t-2}; \theta_{t-1}) \tag{8}$$

where f_{plan} is the planning function, $\hat{p}_t$ represents where the planner expected the ego-vehicle to be at time t, conditioned on the historical context $\mathcal{H}_{t-2}$ available at $t-1$. We then define the temporal planning consistency loss $\mathcal{L}_{temp}$ as the displacement between this prior expectation and the actual ego-state p_t^{obs} derived from onboard sensors:

$$\mathcal{L}_{temp} = \big\|\hat{p}_t - p_t^{obs}\big\|_2^2, \tag{9}$$

By minimizing $\mathcal{L}_{temp}$, we leverage the consistency of downstream planning to provide a supervisory signal for adaptation. This mechanism utilizes the temporal discrepancy between expected and observed states as a cross-task integrity check, ensuring that the adaptive updates remain grounded in the vehicle's actual movement.

4.2 Teacher-Student Architecture for Cross-frame Stability

Autonomous driving is inherently a continuous temporal process where environmental shifts are not isolated events but persistent perturbations. Despite the synergetic guidance imposed by our joint objective $\mathcal{L}_{joint}$, adaptation over continuous test streams faces the risk of error accumulation. Since each update is conditioned on previously adapted states, minor gradient biases at a single frame can gradually compound, leading to severe parameter drift or even catastrophic model collapse.

To address this issue, we introduce a teacher-student stabilization architecture $\mathcal{P}_\Omega$ that strikes a balance between rapid online adaptation and long-horizon stability preservation (Gong et al., 2022;

Döbler et al., 2023; Lee et al., 2023). In this framework, the student model generates candidate parameter updates derived from the current input. Simultaneously, the teacher system constructs a dynamic stability region Ω from both long-term and short-term scales, guiding and ultimately inspecting the student parameter update. To maintain clarity, we denote the teacher's stable anchor as θ^* and the student's states within a single adaptation step as $\theta_{t-1} \to \theta_t^{'} \to \hat{\theta}_t \to \theta_t$ (representing the initial, candidate, clipped, and finalized states, respectively).

Student: Candidate Parameter Update. As the immediate responder to environmental shifts, the student model aims to extract adaptive signals from the current streaming data. At each timestep t, the student computes the joint objective $\mathcal{L}_{joint}$ by integrating the current input x_t with the preserved historical context. Instead of finalizing the update, the student first performs a tentative gradient step to obtain the candidate parameter of the student $\theta_t^{'}$:

$$\theta_t^{'} = \theta_{t-1} - \eta \nabla_\theta \mathcal{L}_{joint}(x_t, x_{t-1}, \mathcal{H}_{t-1}, \mathcal{H}_{t-2}; \theta_{t-1}). \tag{10}$$

These candidate parameters represent a potential adaptation direction that prioritizes the current joint objective. However, the candidate update still requires validation.

Teacher: Two-scale Stability Guidance. While these candidate parameters θ_t' aim to minimize the immediate joint objective, relying solely on per-step optimization can lead to unbounded parameter drift and model instability over long sequences. To address this, we develop a teacher-student guidance mechanism that oversees student updates across two distinct time scales: maintaining a global anchor to suppress long-term cumulative drift, while acting as a short-term gatekeeper to filter destabilizing updates.

Specifically, the teacher model θ^* serves as the global anchor by tracking the reliable history of the student's evolution. Unlike the student, θ^* does not participate in backpropagation; instead, it evolves through an EMA (Tan et al., 2026; Busbridge et al., 2023) of the finalized student parameters:

$$\theta_t^* = \alpha \theta_{t-1}^* + (1-\alpha)\theta_t, \tag{11}$$

where α is a momentum coefficient that controls the update speed of the teacher. By assigning a high value to α, the teacher anchor θ^* evolves much more slowly than the student, effectively filtering out high-frequency gradient noise from transient distribution shift.

In the proposed teacher-student architecture, the teacher anchor serves as the coordinate center for a dynamic stability region. Instead of allowing the student to update freely, the teacher utilizes θ^* as a spatial reference to bound the student's exploration. Specifically, the candidate parameters θ' proposed by the student are forcefully projected into an L_∞ ball of radius r centered at the teacher anchor:

$$\hat{\theta}_t = \Pi_{\mathcal{B}_\infty(\theta_{t-1}^*, r)}(\theta_t^{'}). \tag{12}$$

Here, Π denotes the projection operator that maps the student's candidate $\theta_t^{'}$ back into the L_∞-norm ball $\mathcal{B}_\infty$, which is centered at the previous teacher anchor θ_{t-1}^*. The radius r serves as a critical hyperparameter that defines the trust region of adaptation, determining the maximum allowable deviation between the student's local exploration and the teacher's stable knowledge. By centering this constraint on θ_{t-1}^* instead of θ_{t-1}, the teacher ensures that the student remains anchored to a smooth historical consensus rather than drifting along with potentially noisy frame-to-frame updates. Consequently, this spatial projection effectively tethers the adaptation path, suppressing the long-term cumulative drift by ensuring the parameters remain within a reliable region defined by the teacher.

Complementing the long-term spatial guidance, the teacher simultaneously acts as a short-term gatekeeper to safeguard the system against transient, destabilizing updates. At each timestep, a per-step filter is employed to audit the quality of the candidate update $\hat{\theta}_t$ by evaluating its impact on the model performance.

In the absence of ground-truth labels during online adaptation, we leverage the temporal planning consistency $\mathcal{L}_{temp}$ as a self-supervised proxy to assess this impact. As described in Eq. 9, this metric is grounded in the physical reality of ego-state continuity: the future trajectory predicted by the model at $t-1$ should remain consistent with the actual displacement observed at t. By monitoring this consistency, the teacher can effectively discern whether a candidate update $\hat{\theta}_t$ genuinely improves the model's predictive stability or merely overfits to transient environmental noise. Consequently, the

Table 2: **Planning performance under severe Gaussian noise corruption (severity 5) before and after TTA.** Our method consistently improves performance across the evaluated architectures, recovering 24.7% of the corruption-induced L2 degradation and 26.9% of the collision-rate degradation on average. Blue-shaded rows indicate results after TTA by our method. For each adapted result, red subscripts report the recovery rate, and blue subscripts report the relative change.

Method	L2(m)↓				Col(%)↓			
	1s	2s	3s	AVG	1s	2s	3s	AVG
ST-P3 (Hu et al., 2022)	2.51	5.22	8.13	**5.29**	0.89	5.08	9.42	**5.13**
UniAD (Hu et al., 2023)	0.60	1.29	2.25	**1.38**	0.81	0.83	1.23	**0.95**
VAD-base (Jiang et al., 2023)	0.63	1.04	1.52	**1.06**	0.55	0.73	1.12	**0.80**
DistillDrive (Yu et al., 2025)	0.36	0.75	1.29	**0.80**	0.06	0.24	0.65	**0.31**
SparseDrive (Sun et al., 2025)	0.36	0.73	1.25	**0.78**	0.06	0.27	0.90	**0.41**
DiffusionDrive (Liao et al., 2025)	0.34	0.72	1.25	**0.77**	0.02	0.18	0.68	**0.29**
ST-P3 (Hu et al., 2022)	2.29	4.57	6.95	**4.60**(+34.3%/-13.0%)	0.46	3.88	7.76	**4.03**(+31.3%/-21.4%)
UniAD (Hu et al., 2023)	0.51	1.15	2.09	**1.25**(+29.5%/-9.4%)	0.53	0.57	1.08	**0.72**(+34.8%/-24.2%)
VAD-base (Jiang et al., 2023)	0.56	0.96	1.41	**0.97**(+25.5%/-8.6%)	0.35	0.62	1.06	**0.68**(+29.4%/-15.5%)
DistillDrive (Yu et al., 2025)	0.34	0.71	1.22	**0.76**(+18.2%/-5.0%)	0.06	0.20	0.57	**0.28**(+12.5%/-9.7%)
SparseDrive (Sun et al., 2025)	0.36	0.70	1.16	**0.74**(+26.7%/-5.1%)	0.10	0.22	0.64	**0.32**(+27.3%/-22.0%)
DiffusionDrive (Liao et al., 2025)	0.32	0.68	1.21	**0.74**(+15.0%/-3.9%)	0.06	0.15	0.48	**0.23**(+28.6%/-20.7%)

update is finalized only if it demonstrates a superior alignment with this physical constraint:

$$\theta_t = \begin{cases} \hat{\theta}_t, & \mathcal{L}_{temp}(\cdot;\hat{\theta}_t) < \mathcal{L}_{temp}(\cdot;\theta_{t-1}), \\ \theta_{t-1}, & \text{otherwise.} \end{cases} \tag{13}$$

Through this dual-scale oversight—combining long-term spatial anchoring with short-term behavioral filtering—the teacher system effectively breaks the recursive error chain, ensuring robust and reliable student adaptation across diverse and challenging driving environments.

5 Experiments

5.1 Experimental Setup

As shown in Figure 1(a), Gaussian noise at severity 5 induces the largest performance degradation across all evaluated severity levels on the NuScenes validation set; we therefore adopt this setting as the primary evaluation benchmark throughout. We evaluate the proposed CroTTA across six representative end-to-end driving models, including ST-P3 (Hu et al., 2022), UniAD (Hu et al., 2023), VAD (Jiang et al., 2023), DistillDrive (Yu et al., 2025), SparseDrive (Sun et al., 2025), and DiffusionDrive (Liao et al., 2025), using their officially released pretrained weights. Our approach is compared against five TTA baselines: Tent (Wang et al., 2020), EcoTTA (Song et al., 2023), EATA (Niu et al., 2022), EATA w/o EWC, and SAR (Niu et al., 2023). For all experiments, we employ the Adam optimizer with a learning rate of $\eta = 0.01$. The EMA momentum coefficient and the L_∞ projection radius are set to $\alpha = 0.8$ and $r = 0.005$, respectively, and the loss weight is set to $\lambda = 1$; a detailed sensitivity analysis is available in Appendix D.

5.2 Main Results

We evaluate our method on a diverse set of end-to-end models under severe Gaussian noise corruption; results are reported in Table 2. Across all architectures, our method reduces average L2 by 3.9%–13.0%. More importantly, we observe substantial reductions in collision rate across all architectures, with improvements ranging from 9.7% to 24.2%. Since collision rate directly reflects whether the vehicle can complete a scene without incident, this result suggests that TTA is effective not merely at smoothing predicted trajectories, but at recovering genuinely safer driving behavior under distribution shift. Qualitative trajectories in Figure 2 further illustrate this effect. Across diverse scenarios including intersections and road terminations, and under both severe Gaussian noise and fog corruption, the pretrained model produces unsafe plans such as colliding with surrounding vehicles or drifting onto road shoulders, while our method, applied plug-and-play to each architecture, repairs these failures and recovers plausible trajectories for both straight-ahead and turning maneuvers. More visualizations can be found in Appendix G.

Table 3: **Planning performance on NAVSIM under severity-5 Gaussian noise.** CroTTA improves PDMS for all five planners, recovering 11.8% of the corruption-induced degradation on average. Blue-shaded rows show adapted results; red/blue subscripts denote recovery rate/relative change. NV, DAC, TTC, EP, and PDMS denote no-vehicle-collision, drivable-area compliance, time-to-collision, ego progress, and PDM score, respectively.

Method	NV↑	DAC↑	TTC↑	EP↑	PDMS↑
DiffusionDrive (Liao et al., 2025)	90.4	94.8	80.0	76.7	78.1
WoTE (Li et al., 2025)	95.7	93.3	88.8	73.2	79.7
DriveLaW (Xia et al., 2026)	95.7	91.8	87.2	74.8	79.1
DriveVLA-W0 (Li et al., 2026)	98.6	93.4	94.8	80.8	86.1
GoalFlow (Xing et al., 2025)	97.9	96.7	92.9	83.7	88.1
DiffusionDrive (Liao et al., 2025)	**91.2**(+9.1%/+0.9%)	**95.0**(+16.0%/+0.2%)	**81.2**(+8.1%/+1.5%)	**77.1**(+7.8%/+0.5%)	**78.9**(+8.1%/+1.0%)
WoTE (Li et al., 2025)	**95.9**(+5.4%/+0.2%)	**93.6**(+10.4%/+0.3%)	**89.0**(+2.6%/+0.2%)	**73.5**(+3.3%/+0.4%)	**80.2**(+4.9%/+0.6%)
DriveLaW (Xia et al., 2026)	**95.8**(+2.9%/+0.1%)	**92.2**(+12.5%/+0.4%)	**87.6**(+4.3%/+0.5%)	**75.3**(+10.0%/+0.7%)	**79.8**(+7.1%/+0.9%)
DriveVLA-W0 (Li et al., 2026)	**98.6**(+0.4%/+0.0%)	**93.4**(+2.0%/+0.0%)	**95.4**(+43.0%/+0.6%)	**80.8**(+32.0%/+0.0%)	**86.4**(+30.0%/+0.3%)
GoalFlow (Xing et al., 2025)	**98.0**(+2.3%/+0.1%)	**97.1**(+8.9%/+0.4%)	**93.0**(+10.4%/+0.1%)	**83.8**(+6.9%/+0.1%)	**88.4**(+9.0%/+0.3%)

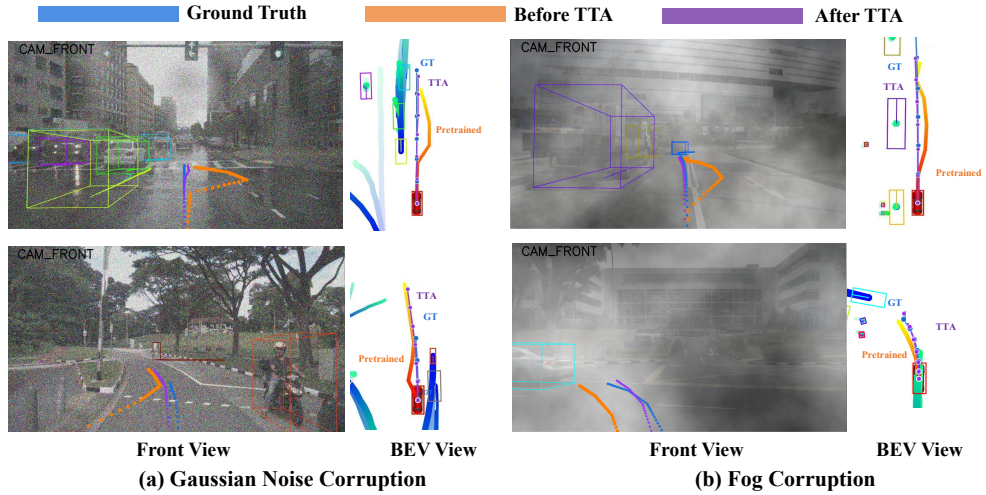


Figure 2: **Qualitative trajectory comparisons before and after TTA** under Gaussian noise and fog corruption across diverse driving scenarios. Our method recovers safe and plausible trajectories where the pretrained model fails.

However, models such as ST-P3 and DiffusionDrive differ in performance under severe disturbance by nearly 7×. We therefore define recovery rate as the ratio of TTA-induced improvement to the total corruption-induced degradation, measuring how much of the distribution-shift damage our method repairs. According to Table 2, recovery rates average 24.7% for L2 and 26.9% for collision rate over the five architectures with positive corruption-induced gaps. The clean model performances on the original NuScenes dataset, which serve as the reference baselines for calculating recovery rates, are detailed in Appendix A.

We also evaluate CroTTA on NAVSIM (Dauner et al., 2024) under severity-5 Gaussian noise. As shown in Table 3, CroTTA improves PDMS for all five planners, with an average recovery rate of 11.8%. The improvements in no-vehicle-collision (NV), drivable-area compliance (DAC), time-to-collision (TTC), ego progress (EP), and PDMS indicate that CroTTA promotes more reasonable and safer driving under distribution shift.

Table 4 further compares our method against representative TTA baselines on UniAD under severe Gaussian noise corruption. Tent and EcoTTA yield negligible gains, with collision rate reductions of at most 2.1%. EATA and SAR improve upon this through sample selection and reliability-aware weighting, reducing L2 by 1.4%–2.9% and collision rate by 6.3%–11.6%. However, these methods were designed for multi-step single-task adaptation. They apply a single surrogate loss, ignoring the signals that flow from perception to planning, and their stability mechanisms assume independent

Table 4: **Comparison with other TTA methods on UniAD.** CroTTA exceeds the strongest competing baseline by 20.5 and 18.2 percentage points in L2 and collision recovery, respectively. Underlined results denote the second-best method.

Method	L2 AVG↓	Col AVG↓
Without TTA	1.38	0.95
Tent (Wang et al., 2020)	**1.37**(+2.3%/-0.7%)	**0.94**(+1.5%/-1.1%)
EcoTTA (Song et al., 2023)	**1.38**(0.0%/0.0%)	**0.93**(+3.0%/-2.1%)
EATA (Niu et al., 2022)	**1.35**(+6.8%/-2.2%)	**0.84**(+16.7%/-11.6%)
EATA–E (Niu et al., 2022)	**1.34**(+9.1%/-2.9%)	**0.89**(+9.1%/-6.3%)
SAR (Niu et al., 2023)	**1.36**(+4.5%/-1.4%)	**0.86**(+13.6%/-9.5%)
CroTTA (Ours)	**1.25**(+29.5%/-9.4%)	**0.72**(+34.8%/-24.2%)

Table 5: **Ablation study of key components on UniAD.** The full method achieves 29.5% L2 and 34.8% collision recovery. $\mathcal{L}_{mm}$, $\mathcal{L}_{temp}$, and $\mathcal{P}_{\Omega}$ denote perception entropy, temporal consistency, and teacher–student stabilization.

$\mathcal{L}_{mm}$	$\mathcal{L}_{temp}$	$\mathcal{P}_{\Omega}$	L2 AVG↓	Col AVG↓
-	-	-	1.38	0.95
✓	-	-	**1.33**(+11.4%/-3.6%)	**0.92**(+4.5%/-3.2%)
-	✓	-	**1.32**(+13.6%/-4.3%)	**0.92**(+4.5%/-3.2%)
✓	✓	-	**1.32**(+13.6%/-4.3%)	**0.81**(+21.2%/-14.7%)
✓	✓	✓	**1.25**(+29.5%/-9.4%)	**0.72**(+34.8%/-24.2%)

test samples, offering insufficient protection against error accumulation over a continuously evolving driving sequence. However, by jointly coordinating perception and planning objectives while anchoring updates across frames via the teacher model, our method achieves the best performance across both metrics, yielding nearly 3× greater L2 improvement and over 2× the collision rate reduction compared to the second-best baseline.

5.3 Ablation and Efficiency Analysis

To isolate the effect of each component, we evaluate variants of our method on UniAD under severe Gaussian noise disturbance, progressively adding $\mathcal{L}_{mm}$, $\mathcal{L}_{temp}$, and the teacher-student architecture $\mathcal{P}_{\Omega}$; results are summarized in Table 5.

Either loss alone recovers 11.4%–13.6% of the L2 degradation but only 4.5% of the collision-rate degradation. Combining $\mathcal{L}_{mm}$ and $\mathcal{L}_{temp}$ raises collision recovery to 21.2%, a gain of 16.7 percentage points over either objective alone, while matching temporal adaptation in L2. Thus, coordinating the adaptation signals improves a planning outcome that remains largely unrecovered with either isolated objective. Adding teacher–student stabilization raises collision recovery by a further 13.6 percentage points to 34.8%, while the full method achieves 29.5% L2 recovery. Appendix E examines these contributions on ST-P3 and SparseDrive.

Table 6 evaluates the overhead of online adaptation. Across six models, CroTTA updates fewer than 0.15‰ of parameters and requires under 20ms per frame on average, supporting recovery within a small additional computational budget. This lightweight update recovers 26.9% of the collision-rate degradation on average. Even UniAD, which has the highest adaptation cost, requires only 47.21ms per frame, supporting the practical use of CroTTA across the evaluated architectures.

Table 6: **Parameter and TTA time efficiency.** CroTTA updates <0.23‰ parameters and stays under 50ms per frame for all models.

Model	Parameters Ratio (‰)	Time (ms) Mean	Std
UniAD (Hu et al., 2023)	0.229	47.2	18.3
VAD (Jiang et al., 2023)	0.162	10.6	6.7
ST-P3 (Hu et al., 2022)	0.109	13.7	8.1
DiffusionDrive (Liao et al., 2025)	0.095	16.4	18.4
SparseDrive (Sun et al., 2025)	0.089	17.7	8.5
DistillDrive (Yu et al., 2025)	0.078	16.4	13.1

6 Conclusion

In this paper, we presented CroTTA, a plug-and-play framework designed for cross-task and cross-frame TTA of end-to-end autonomous driving models. By moving beyond isolated task updates, CroTTA introduces a joint adaptation objective that harmonizes upstream perception uncertainty with downstream planning consistency, effectively mitigating task-level interference through cross-task coordination. To ensure reliability over long horizons, we developed a teacher-student architecture that suppresses cumulative parameter drift and prevents model collapse via multi-level anchoring and update verification. Extensive evaluations across diverse driving architectures demonstrate that CroTTA effectively mitigates severe distribution shifts, achieving substantial reductions in displacement errors and collision rates while maintaining minimal computational overhead.

AI USE STATEMENT

Generative AI tools were used to assist with manuscript editing, including language polishing, formatting, and consistency checks. They were not used to generate experimental data or results. All AI-assisted content was reviewed and verified by the authors, who take responsibility for the final manuscript.

ETHICS STATEMENT

This work studies the robustness of autonomous-driving models using public benchmarks and simulated corruptions. It involves no human-subject experiments or deployment on public roads; any real-world use would require independent safety validation before deployment.

REPRODUCIBILITY STATEMENT

The experimental setup, hyperparameter analyses, and baseline implementation details are provided in the main paper and Appendices D and F. Our anonymous repository at `https://anonymous.4open.science/r/CroTTA-88D0/` currently provides the UniAD implementation, corruption-generation pipeline, and evaluation instructions; implementations and standardized configurations for the remaining evaluated architectures will be added progressively.

REFERENCES

Shai Ben-David, John Blitzer, Koby Crammer, and Fernando Pereira. Analysis of representations for domain adaptation. *Advances in neural information processing systems*, 19, 2006.

Dan Busbridge, Jason Ramapuram, Pierre Ablin, Tatiana Likhomanenko, Eeshan Gunesh Dhekane, Xavier Suau Cuadros, and Russell Webb. How to scale your ema. *Advances in Neural Information Processing Systems*, 36:73122–73174, 2023.

Dian Chen, Dequan Wang, Trevor Darrell, and Sayna Ebrahimi. Contrastive test-time adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 295–305, 2022.

Guohao Chen, Shuaicheng Niu, Deyu Chen, Shuhai Zhang, Changsheng Li, Yuanqing Li, and Mingkui Tan. Cross-device collaborative test-time adaptation. *Advances in Neural Information Processing Systems*, 37:122917–122951, 2024a.

Li Chen, Penghao Wu, Kashyap Chitta, Bernhard Jaeger, Andreas Geiger, and Hongyang Li. End-to-end autonomous driving: Challenges and frontiers. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 46(12):10164–10183, 2024b.

Zhao Chen, Vijay Badrinarayanan, Chen-Yu Lee, and Andrew Rabinovich. Gradnorm: Gradient normalization for adaptive loss balancing in deep multitask networks. In *International conference on machine learning*, pp. 794–803. PMLR, 2018.

Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *IEEE transactions on pattern analysis and machine intelligence*, 45(11):12878–12895, 2022.

Shuyang Chu, Jingang Shi, Xu Cheng, Haoyu Chen, Xin Liu, Jian Xu, and Guoying Zhao. To remember, to adapt, to preempt: A stable continual test-time adaptation framework for remote physiological measurement in dynamic domain shifts. In *Proceedings of the 33rd ACM International Conference on Multimedia*, pp. 7307–7316, 2025.

Felipe Codevilla, Antonio M Lopez, Vladlen Koltun, and Alexey Dosovitskiy. On offline evaluation of vision-based driving models. In *Proceedings of the European conference on computer vision (ECCV)*, pp. 236–251, 2018.

Felipe Codevilla, Eder Santana, Antonio M López, and Adrien Gaidon. Exploring the limitations of behavior cloning for autonomous driving. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 9329–9338, 2019.

Nicolas Courty, Rémi Flamary, Devis Tuia, and Alain Rakotomamonjy. Optimal transport for domain adaptation. *IEEE transactions on pattern analysis and machine intelligence*, 39(9):1853–1865, 2016.

Karan Dalal, Daniel Koceja, Jiarui Xu, Yue Zhao, Shihao Han, Ka Chun Cheung, Jan Kautz, Yejin Choi, Yu Sun, and Xiaolong Wang. One-minute video generation with test-time training. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 17702–17711, 2025.

Daniel Dauner, Marcel Hallgarten, Tianyu Li, Xinshuo Weng, Zhiyu Huang, Zetong Yang, Hongyang Li, Igor Gilitschenski, Boris Ivanovic, Marco Pavone, Andreas Geiger, and Kashyap Chitta. NAVSIM: Data-driven non-reactive autonomous vehicle simulation and benchmarking. In *Advances in Neural Information Processing Systems*, volume 37, pp. 28706–28719, 2024.

Weijian Deng, Stephen Gould, and Liang Zheng. What does rotation prediction tell us about classifier accuracy under varying testing environments? In *International Conference on Machine Learning*, pp. 2579–2589. PMLR, 2021.

Christopher Diehl, Peter Karkus, Sushant Veer, Marco Pavone, and Torsten Bertram. Lord: Adapting differentiable driving policies to distribution shifts. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 7036–7043. IEEE, 2025.

Tuong Do, Binh X Nguyen, Quang D Tran, Hien Nguyen, Erman Tjiputra, Te-Chuan Chiu, and Anh Nguyen. Reducing non-iid effects in federated autonomous driving with contrastive divergence loss. In *2024 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 2190–2196. IEEE, 2024.

Mario Döbler, Robert A Marsden, and Bin Yang. Robust mean teacher for continual and gradual test-time adaptation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 7704–7714, 2023.

Abolfazl Farahani, Sahar Voghoei, Khaled Rasheed, and Hamid R Arabnia. A brief review of domain adaptation. *Advances in data science and information engineering: proceedings from ICDATA 2020 and IKE 2020*, pp. 877–894, 2021.

Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario March, and Victor Lempitsky. Domain-adversarial training of neural networks. *Journal of machine learning research*, 17(59):1–35, 2016.

Taesik Gong, Jongheon Jeong, Taewon Kim, Yewon Kim, Jinwoo Shin, and Sung-Ju Lee. Note: Robust continual test-time adaptation against temporal correlation. *Advances in Neural Information Processing Systems*, 35:27253–27266, 2022.

Sorin Grigorescu, Bogdan Trasnea, Tiberiu Cocias, and Gigel Macesanu. A survey of deep learning techniques for autonomous driving. *Journal of field robotics*, 37(3):362–386, 2020.

Yongping He, Yuanqing Xia, Tijin Yan, Yufeng Zhan, Jiaru Song, and Zihang Feng. Unsupervised cross-domain attack traffic classifier for intelligent connected vehicle. *IEEE Transactions on Industrial Electronics*, 2025.

Dan Hendrycks and Thomas Dietterich. Benchmarking neural network robustness to common corruptions and perturbations. *arXiv preprint arXiv:1903.12261*, 2019.

Trung-Hieu Hoang, Duc M Vo, et al. Persistent test-time adaptation in recurring testing scenarios. *Advances in Neural Information Processing Systems*, 37:123402–123442, 2024.

Shengchao Hu, Li Chen, Penghao Wu, Hongyang Li, Junchi Yan, and Dacheng Tao. St-p3: End-to-end vision-based autonomous driving via spatial-temporal feature learning. In *European Conference on Computer Vision*, pp. 533–549. Springer, 2022.

Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 17853–17862, 2023.

Zhiyu Huang, Haochen Liu, and Chen Lv. Gameformer: Game-theoretic modeling and learning of transformer-based interactive prediction and planning for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3903–3913, 2023.

Bo Jiang, Shaoyu Chen, Qing Xu, Bencheng Liao, Jiajie Chen, Helong Zhou, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. Vad: Vectorized scene representation for efficient autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 8340–8350, 2023.

Xiyan Jiang, Xiaocong Zhao, Yiru Liu, Zirui Li, Peng Hang, Lu Xiong, and Jian Sun. A naturalistic trajectory dataset with dense interaction for autonomous driving. *Scientific Data*, 12(1):1084, 2025. doi: 10.1038/s41597-025-05344-7.

Adilbek Karmanov, Dayan Guan, Shijian Lu, Abdulmotaleb El Saddik, and Eric Xing. Efficient test-time adaptation of vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14162–14171, 2024.

Elyor Kodirov, Tao Xiang, Zhenyong Fu, and Shaogang Gong. Unsupervised domain adaptation for zero-shot learning. In *Proceedings of the IEEE international conference on computer vision*, pp. 2452–2460, 2015.

Taeyeop Lee, Jonathan Tremblay, Valts Blukis, Bowen Wen, Byeong-Uk Lee, Inkyu Shin, Stan Birchfield, In So Kweon, and Kuk-Jin Yoon. Tta-cope: Test-time adaptation for category-level object pose estimation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 21285–21295, 2023.

Yingyan Li, Yuqi Wang, Yang Liu, Jiawei He, Lue Fan, and Zhaoxiang Zhang. End-to-end driving with online trajectory evaluation via BEV world model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 27137–27146, 2025.

Yingyan Li, Shuyao Shang, Weisong Liu, Bing Zhan, Haochen Wang, Yuqi Wang, Yuntao Chen, Xiaoman Wang, Yasong An, Chufeng Tang, Lu Hou, Lue Fan, and Zhaoxiang Zhang. DriveVLA-W0: World models amplify data scaling law in autonomous driving. In *International Conference on Learning Representations*, 2026.

Jian Liang, Ran He, and Tieniu Tan. A comprehensive survey on test-time adaptation under distribution shifts. *International Journal of Computer Vision*, 133(1):31–64, 2025.

Bencheng Liao, Shaoyu Chen, Haoran Yin, Bo Jiang, Cheng Wang, Sixu Yan, Xinbang Zhang, Xiangyu Li, Ying Zhang, Qian Zhang, et al. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 12037–12047, 2025.

Andrey Malinin, Neil Band, German Chesnokov, Yarin Gal, Mark JF Gales, Alexey Noskov, Andrey Ploskonosov, Liudmila Prokhorenkova, Ivan Provilkov, Vatsal Raina, et al. Shifts: A dataset of real distributional shift across multiple large-scale tasks. *arXiv preprint arXiv:2107.07455*, 2021.

Liangbo Ning, Zuowei Zhang, Weiping Ding, Dian Shao, and Yining Zhu. Multilevel distribution alignment for multisource universal domain adaptation. *IEEE Transactions on Neural Networks and Learning Systems*, 2025.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Shijian Zheng, Peilin Zhao, and Mingkui Tan. Efficient test-time model adaptation without forgetting. In *International conference on machine learning*, pp. 16888–16905. PMLR, 2022.

Shuaicheng Niu, Jiaxiang Wu, Yifan Zhang, Zhiquan Wen, Yaofo Chen, Peilin Zhao, and Mingkui Tan. Towards stable test-time adaptation in dynamic wild world. *arXiv preprint arXiv:2302.12400*, 2023.

David Osowiechi, Gustavo A Vargas Hakim, Mehrdad Noori, Milad Cheraghalikhani, Ali Bahri, Moslem Yazdanpanah, Ismail Ben Ayed, and Christian Desrosiers. Nc-ttt: A noise constrastive approach for test-time training. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 6078–6086, 2024.

Yaniv Ovadia, Emily Fertig, Jie Ren, Zachary Nado, David Sculley, Sebastian Nowozin, Joshua Dillon, Balaji Lakshminarayanan, and Jasper Snoek. Can you trust your model's uncertainty? evaluating predictive uncertainty under dataset shift. *Advances in neural information processing systems*, 32, 2019.

Hyewon Park, Hyejin Park, Jueun Ko, and Dongbo Min. Hybrid-tta: Continual test-time adaptation via dynamic domain shift detection. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2877–2886, 2025.

Xingchao Peng, Qinxun Bai, Xide Xia, Zijun Huang, Kate Saenko, and Bo Wang. Moment matching for multi-source domain adaptation. In *Proceedings of the IEEE/CVF international conference on computer vision*, pp. 1406–1415, 2019.

Taki Hasan Rafi, Amit Agarwal, Hitesh L Patel, and Dong-Kyu Chae. Learning from unknown for open-set test-time adaptation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2993–3004, 2026.

Jin Shin and Hyun Kim. L-tta: Lightweight test-time adaptation using a versatile stem layer. *Advances in Neural Information Processing Systems*, 37:39325–39349, 2024.

Youssef Shoeb, Azarm Nowzad, and Hanno Gottschalk. Out-of-distribution segmentation in autonomous driving: Problems and state of the art. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 4310–4320, 2025.

Kihyuk Sohn, Sifei Liu, Guangyu Zhong, Xiang Yu, Ming-Hsuan Yang, and Manmohan Chandraker. Unsupervised domain adaptation for face recognition in unlabeled videos. In *Proceedings of the IEEE International Conference on computer vision*, pp. 3210–3218, 2017.

Damian Sójka, Sebastian Cygert, Bartłomiej Twardowski, and Tomasz Trzciński. Ar-tta: A simple method for real-world continual test-time adaptation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 3491–3495, 2023.

Junha Song, Jungsoo Lee, In So Kweon, and Sungha Choi. Ecotta: Memory-efficient continual test-time adaptation via self-distilled regularization. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 11920–11929, 2023.

Wenchao Sun, Xuewu Lin, Yining Shi, Chuang Zhang, Haoran Wu, and Sifa Zheng. Sparsedrive: End-to-end autonomous driving via sparse scene representation. In *2025 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 8795–8801. IEEE, 2025.

Yu Sun, Xiaolong Wang, Zhuang Liu, John Miller, Alexei Efros, and Moritz Hardt. Test-time training with self-supervision for generalization under distribution shifts. In *International conference on machine learning*, pp. 9229–9248. PMLR, 2020.

Jiayao Tan, Fan Lyu, Chenggong Ni, Fuyuan Hu, Wei Feng, and Rui Yao. Exposing mixture and annotating confusion for active universal test-time adaptation. In *The Fourteenth International Conference on Learning Representations*, 2026.

Mingkui Tan, Guohao Chen, Jiaxiang Wu, Yifan Zhang, Yaofo Chen, Peilin Zhao, and Shuaicheng Niu. Uncertainty-calibrated test-time model adaptation without forgetting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2025.

Hui Tang and Kui Jia. Discriminative adversarial domain adaptation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 5940–5947, 2020.

Dequan Wang, Evan Shelhamer, Shaoteng Liu, Bruno Olshausen, and Trevor Darrell. Tent: Fully test-time adaptation by entropy minimization. *arXiv preprint arXiv:2006.10726*, 2020.

Haotao Wang, Chaowei Xiao, Jean Kossaifi, Zhiding Yu, Anima Anandkumar, and Zhangyang Wang. Augmax: Adversarial composition of random augmentations for robust training. *Advances in neural information processing systems*, 34:237–250, 2021.

Qin Wang, Olga Fink, Luc Van Gool, and Dengxin Dai. Continual test-time domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 7201–7211, 2022.

Tianze Xia, Yongkang Li, Lijun Zhou, Jingfeng Yao, Kaixin Xiong, Haiyang Sun, Bing Wang, Kun Ma, Guang Chen, Hangjun Ye, Wenyu Liu, and Xinggang Wang. DriveLaW: Unifying planning and video generation in a latent driving world. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 39701–39712, 2026.

Tete Xiao, Xiaolong Wang, Alexei A Efros, and Trevor Darrell. What should not be contrastive in contrastive learning. *arXiv preprint arXiv:2008.05659*, 2020.

Zehao Xiao and Cees GM Snoek. Beyond model adaptation at test time: A survey. *arXiv preprint arXiv:2411.03687*, 2024.

Zebin Xing, Xingyu Zhang, Yang Hu, Bo Jiang, Tong He, Qian Zhang, Xiaoxiao Long, and Wei Yin. GoalFlow: Goal-driven flow matching for multimodal trajectories generation in end-to-end autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 1602–1611, 2025.

Yanyu Ye, Wei Wei, Lei Zhang, Chen Ding, and Yanning Zhang. Domain consistency learning for continual test-time adaptation in image semantic segmentation. *Pattern Recognition*, 165:111585, 2025.

Kaichao You, Mingsheng Long, Zhangjie Cao, Jianmin Wang, and Michael I Jordan. Universal domain adaptation. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 2720–2729, 2019.

Rui Yu, Xianghang Zhang, Runkai Zhao, Huaicheng Yan, and Meng Wang. Distilldrive: End-to-end multi-mode autonomous driving distillation by isomorphic hetero-source planning model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 26188–26197, 2025.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. *Advances in neural information processing systems*, 33: 5824–5836, 2020.

Yizhe Zhang, Shubhankar Borse, Hong Cai, and Fatih Porikli. Auxadapt: Stable and efficient test-time adaptation for temporally consistent video semantic segmentation. In *Proceedings of the IEEE/CVF winter conference on applications of computer vision*, pp. 2339–2348, 2022.

Bowen Zhao, Chen Chen, and Shu-Tao Xia. Delta: degradation-free fully test-time adaptation. *arXiv preprint arXiv:2301.13018*, 2023.

Seth Z Zhao, Luobin Wang, Hongwei Ruan, Yuxin Bao, Yilan Chen, Ziyang Leng, Abhijit Ravichandran, Honglin He, Zewei Zhou, Xu Han, et al. Bridgesim: Unveiling the ol-cl gap in end-to-end autonomous driving. *arXiv preprint arXiv:2604.10856*, 2026.

Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *Proceedings of the AAAI conference on artificial intelligence*, volume 34, pp. 13001–13008, 2020.

A Supplementary Experiments on Clean Data and Diverse Visual Corruptions

The main experiments evaluate CroTTA under severe Gaussian noise. Here, we assess adaptation on clean inputs and under fog, brightness, and snow, then extend the evaluation to contrast corruption whose severity changes over time.

A.1 Clean Data and Recovery-Rate Definition

Adaptation should remain useful when inputs are clean as well as when they are corrupted. We therefore evaluate CroTTA on the original nuScenes validation sequences. The unadapted clean results also provide the reference for measuring how much corruption-induced degradation is recovered. For a metric where lower is better, we define the recovery rate as

$$RR = \frac{M_{corrupted} - M_{adapted}}{M_{corrupted} - M_{clean}} \times 100\%. \tag{14}$$

Here, M_{clean} and $M_{corrupted}$ denote the unadapted model's scores on clean and corrupted inputs, and $M_{adapted}$ denotes its score after adaptation on corrupted inputs. Recovery rates are computed for positive corruption-induced gaps; values above 100% mean that adaptation surpasses the clean reference. Mean recovery rates below average the valid rates for each metric. Table notation follows Table 2: blue shading marks adapted results, and red/blue percentages report recovery rate/relative change.

Table 7 shows mean relative reductions of 3.2% in L2 and 5.3% in collision rate across six models. Thus, the gains under corruption are accompanied by modest improvements on clean data, supporting the use of the same adaptation objective in both conditions.

Table 7: **Planning performance on clean nuScenes data.** CroTTA reduces L2 error by 3.2% and collision rate by 5.3% on average across six models, without synthetic corruption.

Method	L2(m)↓				Col(%)↓			
	1s	2s	3s	AVG	1s	2s	3s	AVG
ST-P3 (Hu et al., 2022)	1.72	3.26	4.86	**3.28**	0.42	1.16	3.26	**1.61**
VAD (Jiang et al., 2023)	0.65	1.12	1.65	**1.14**	0.51	0.76	1.34	**0.87**
UniAD (Hu et al., 2023)	0.44	0.88	1.51	**0.94**	0.13	0.20	0.53	**0.29**
SparseDrive (Sun et al., 2025)	0.31	0.60	0.99	**0.63**	0.01	0.05	0.18	**0.08**
DistillDrive (Yu et al., 2025)	0.28	0.55	0.91	**0.58**	0.00	0.03	0.18	**0.07**
DiffusionDrive (Liao et al., 2025)	0.27	0.54	0.90	**0.57**	0.03	0.04	0.16	**0.08**
ST-P3 (Hu et al., 2022)	1.64	3.11	4.67	**3.14**(-4.4%)	0.31	1.20	3.26	**1.59**(-1.2%)
VAD (Jiang et al., 2023)	0.59	1.04	1.56	**1.06**(-7.0%)	0.51	0.74	1.26	**0.84**(-3.7%)
UniAD (Hu et al., 2023)	0.43	0.87	1.51	**0.94**(-0.6%)	0.13	0.20	0.51	**0.28**(-4.2%)
SparseDrive (Sun et al., 2025)	0.31	0.58	0.98	**0.61**(-2.8%)	0.00	0.04	0.20	**0.08**(-7.5%)
DistillDrive (Yu et al., 2025)	0.28	0.54	0.90	**0.57**(-2.8%)	0.00	0.03	0.18	**0.07**(-1.4%)
DiffusionDrive (Liao et al., 2025)	0.26	0.53	0.88	**0.56**(-1.7%)	0.02	0.04	0.14	**0.07**(-14.3%)

A.2 Fog Corruption

Fog tests adaptation when reduced visibility obscures scene structure. Under severity-5 fog, Table 8 shows mean recovery rates of 67.2% for L2 and 28.8% for collision rate. The corresponding relative reductions across all six architectures average 9.5% and 10.7%. These results show that CroTTA can recover planning performance when degradation arises from reduced visibility as well as Gaussian noise.

A.3 Brightness Corruption

Brightness corruption tests whether CroTTA remains effective under changes in illumination. Table 9 reports mean recovery rates of 115.6% for L2 and 38.7% for collision rate, with mean relative reductions of 7.5% and 14.2% across all six architectures. The improvement under illumination shifts extends the benefit to visual changes that do not involve weather-induced occlusion.

A.4 Snow Corruption

Snow introduces weather-induced occlusion and produces the largest average degradation among fog, brightness, and snow in our evaluation. Despite this stronger disturbance, Table 10 shows mean recovery rates of 23.0% for L2 and 53.3% for collision rate. Relative reductions across all six

Table 8: **Planning performance under severe fog corruption (severity 5).** CroTTA recovers 67.2% of L2 degradation and 28.8% of collision-rate degradation on average over models with positive gaps.

Method	L2(m)↓				Col(%)↓			
	1s	2s	3s	AVG	1s	2s	3s	AVG
ST-P3 (Hu et al., 2022)	2.75	5.74	8.81	**5.77**	0.38	5.00	10.77	**5.38**
VAD (Jiang et al., 2023)	0.72	1.22	1.80	**1.25**	0.54	0.84	1.46	**0.95**
UniAD (Hu et al., 2023)	0.47	0.96	1.66	**1.03**	0.17	0.32	0.47	**0.32**
SparseDrive (Sun et al., 2025)	0.31	0.60	0.98	**0.63**	0.00	0.07	0.28	**0.12**
DistillDrive (Yu et al., 2025)	0.30	0.60	0.99	**0.63**	0.03	0.08	0.25	**0.12**
DiffusionDrive (Liao et al., 2025)	0.29	0.58	0.98	**0.62**	0.02	0.09	0.25	**0.12**
ST-P3 (Hu et al., 2022)	1.98	3.77	5.58	**3.78**(+79.9%/-34.5%)	0.64	2.70	6.12	**3.15**(+59.2%/-41.4%)
VAD (Jiang et al., 2023)	0.64	1.11	1.66	**1.14**(+102.8%/-8.8%)	0.53	0.81	1.42	**0.92**(+35.5%/-2.8%)
UniAD (Hu et al., 2023)	0.45	0.92	1.62	**1.00**(+33.3%/-2.9%)	0.17	0.31	0.45	**0.31**(+33.3%/-3.1%)
SparseDrive (Sun et al., 2025)	0.30	0.59	0.97	**0.62**(–/-1.6%)	0.02	0.06	0.26	**0.11**(+25.0%/-8.3%)
DistillDrive (Yu et al., 2025)	0.29	0.58	0.97	**0.61**(+40.0%/-3.2%)	0.02	0.07	0.25	**0.11**(+20.0%/-8.3%)
DiffusionDrive (Liao et al., 2025)	0.27	0.55	0.92	**0.58**(+80.0%/-6.5%)	0.02	0.09	0.24	**0.12**(0.0%/0.0%)

Table 9: **Planning performance under brightness corruption.** CroTTA improves planning under illumination shifts, with mean relative reductions of 7.5% in L2 and 14.2% in collision rate across six architectures.

Method	L2(m)↓				Col(%)↓			
	1s	2s	3s	AVG	1s	2s	3s	AVG
ST-P3 (Hu et al., 2022)	2.10	4.00	6.05	**4.06**	0.62	2.01	4.63	**2.42**
UniAD (Hu et al., 2023)	0.46	0.91	1.57	**0.98**	0.23	0.30	0.45	**0.33**
VAD (Jiang et al., 2023)	0.68	1.17	1.72	**1.19**	0.40	0.72	1.19	**0.77**
DistillDrive (Yu et al., 2025)	0.28	0.56	0.95	**0.60**	0.04	0.09	0.23	**0.12**
SparseDrive (Sun et al., 2025)	0.39	0.77	1.30	**0.82**	0.00	0.20	0.42	**0.20**
DiffusionDrive (Liao et al., 2025)	0.36	0.71	1.20	**0.75**	0.00	0.18	0.42	**0.20**
ST-P3 (Hu et al., 2022)	1.95	3.73	5.61	**3.76**(+38.5%/-7.4%)	0.58	1.54	4.23	**2.11**(+38.3%/-12.8%)
UniAD (Hu et al., 2023)	0.44	0.89	1.55	**0.97**(+34.0%/-1.4%)	0.24	0.28	0.44	**0.32**(+18.2%/-2.0%)
VAD (Jiang et al., 2023)	0.49	0.88	1.38	**0.92**(+546.7%/-23.0%)	0.43	0.71	1.16	**0.77**(–/-0.4%)
DistillDrive (Yu et al., 2025)	0.27	0.56	0.95	**0.60**(+21.6%/-0.7%)	0.00	0.05	0.21	**0.09**(+62.5%/-25.4%)
SparseDrive (Sun et al., 2025)	0.35	0.70	1.19	**0.75**(+36.6%/-8.4%)	0.00	0.18	0.36	**0.18**(+16.5%/-10.0%)
DiffusionDrive (Liao et al., 2025)	0.34	0.68	1.15	**0.72**(+16.3%/-4.0%)	0.00	0.09	0.30	**0.13**(+57.9%/-34.8%)

architectures average 10.4% and 19.4%, respectively. Together, these three corruption tests show that CroTTA improves planning across visibility, illumination, and occlusion shifts using the same adaptation objective.

A.5 Temporally Evolving Corruption

Visual conditions can change within a driving sequence. To evaluate adaptation in this setting, we vary contrast-corruption severity over 20 frames, increasing it during the first 10 frames and decreasing it during the next 10, while retaining adapted parameters across consecutive frames. Table 11 shows lower average L2 error and collision rate for all six architectures, with mean relative reductions of 11.9% and 15.9%, respectively. These sequence-level gains show that adaptation remains useful when the disturbance changes during inference.

Table 10: **Planning performance under snow corruption.** CroTTA remains effective under the strongest of the three additional weather and illumination disturbances, recovering 23.0% of L2 degradation and 53.3% of collision-rate degradation on average over models with positive gaps.

Method	L2(m)↓				Col(%)↓			
	1s	2s	3s	AVG	1s	2s	3s	AVG
ST-P3 (Hu et al., 2022)	3.67	7.67	11.72	**7.69**	2.88	8.23	8.64	**6.58**
UniAD (Hu et al., 2023)	0.44	1.10	2.00	**1.18**	0.38	1.13	1.41	**0.97**
VAD (Jiang et al., 2023)	0.56	0.97	1.48	**1.00**	0.18	0.91	2.11	**1.07**
DistillDrive (Yu et al., 2025)	0.46	0.92	1.53	**0.97**	0.19	0.53	0.58	**0.44**
SparseDrive (Sun et al., 2025)	0.42	0.81	1.33	**0.85**	0.19	0.44	0.84	**0.49**
DiffusionDrive (Liao et al., 2025)	0.47	0.98	1.65	**1.03**	0.04	0.22	0.95	**0.40**
ST-P3 (Hu et al., 2022)	2.97	6.54	10.12	**6.54**(+26.1%/-15.0%)	1.03	3.09	6.17	**3.43**(+63.4%/-47.9%)
UniAD (Hu et al., 2023)	0.42	1.04	1.93	**1.13**(+20.8%/-4.2%)	0.48	0.68	1.34	**0.84**(+19.1%/-13.4%)
VAD (Jiang et al., 2023)	0.43	0.77	1.27	**0.82**(–/-17.9%)	0.00	0.45	1.45	**0.63**(+220.3%/-40.6%)
DistillDrive (Yu et al., 2025)	0.40	0.82	1.38	**0.87**(+25.6%/-10.3%)	0.19	0.48	0.58	**0.42**(+5.4%/-4.5%)
SparseDrive (Sun et al., 2025)	0.38	0.75	1.24	**0.80**(+22.7%/-5.9%)	0.09	0.43	0.90	**0.48**(+2.4%/-2.0%)
DiffusionDrive (Liao et al., 2025)	0.42	0.88	1.51	**0.94**(+19.6%/-8.7%)	0.07	0.23	0.80	**0.37**(+9.4%/-7.5%)

Table 11: **Planning performance under temporally evolving contrast corruption.** With updates retained as severity increases and decreases, CroTTA reduces L2 error by 11.9% and collision rate by 15.9% on average across six architectures.

Method	L2(m)↓				Col(%)↓			
	1s	2s	3s	AVG	1s	2s	3s	AVG
ST-P3 (Hu et al., 2022)	3.47	6.72	10.28	**6.82**	0.41	4.94	9.47	**4.94**
UniAD (Hu et al., 2023)	0.55	1.15	2.01	**1.24**	0.56	0.48	0.78	**0.61**
VAD (Jiang et al., 2023)	0.71	1.22	1.80	**1.24**	0.36	0.71	1.21	**0.76**
DistillDrive (Yu et al., 2025)	0.37	0.77	1.31	**0.82**	0.01	0.09	0.49	**0.20**
SparseDrive (Sun et al., 2025)	0.38	0.78	1.33	**0.83**	0.00	0.08	0.42	**0.17**
DiffusionDrive (Liao et al., 2025)	0.38	0.78	1.33	**0.83**	0.01	0.09	0.49	**0.20**
ST-P3 (Hu et al., 2022)	2.76	5.58	8.60	**5.65**(+33.2%/-17.2%)	0.21	2.67	5.76	**2.88**(+61.9%/-41.7%)
UniAD (Hu et al., 2023)	0.50	1.05	1.87	**1.14**(+32.6%/-7.8%)	0.41	0.56	0.68	**0.55**(+17.9%/-9.3%)
VAD (Jiang et al., 2023)	0.50	0.91	1.44	**0.95**(+283.9%/-23.6%)	0.21	0.52	0.99	**0.57**(+169.7%/-24.6%)
DistillDrive (Yu et al., 2025)	0.34	0.72	1.25	**0.77**(+19.7%/-5.7%)	0.02	0.10	0.43	**0.18**(+10.5%/-6.8%)
SparseDrive (Sun et al., 2025)	0.34	0.70	1.20	**0.75**(+41.7%/-10.0%)	0.02	0.09	0.36	**0.16**(+11.5%/-6.0%)
DiffusionDrive (Liao et al., 2025)	0.34	0.73	1.25	**0.77**(+21.8%/-6.8%)	0.02	0.10	0.43	**0.18**(+11.4%/-6.8%)

B Concurrent Localization and Visual Perturbations

Beyond camera degradation, driving models must also contend with errors in localization signals. Since CroTTA uses the observed ego-state to constrain temporal planning consistency, we jointly perturb GPS position and yaw measurements and apply motion blur. This experiment tests adaptation when both image inputs and the localization reference are noisy.

B.1 Perturbation Severity on UniAD

We first evaluate UniAD at five severity levels. GPS errors of 0.2, 0.5, 1, 2, and 4 m are paired with yaw errors of 0.5°, 1°, 2°, 5°, and 10°, respectively; motion blur is generated with `imagecorruptions`. Table 12 shows recovery in both metrics at severities 1–3, including 57.5%–72.1% L2 recovery. Thus, an imperfect localization reference can still support effective adaptation under moderate joint disturbance.

At severity 4, L2 recovery falls to 3.9%; at severity 5, it becomes −0.3%, although collision recovery remains 52.4%. A 4 m position error is on the scale of a lane width. Together with motion blur and yaw error, this severely corrupts both the prediction and its reference, which is consistent with the failure to obtain a reliable temporal signal for restoring trajectory accuracy.

B.2 Evaluation across Architectures

We next evaluate all six architectures at severity 3, a moderate setting with a 1 m GPS error, a 2° yaw error, and motion blur. Table 13 shows mean recovery rates of 17.9% for L2 and 36.5% for

Table 12: **UniAD under concurrent localization noise and motion blur.** CroTTA recovers 57.5%–72.1% of L2 degradation at severities 1–3, but cannot recover trajectory accuracy at severity 5.

Severity (GPS/yaw)	L2(m)↓ 1s	2s	3s	AVG	Col(%)↓ 1s	2s	3s	AVG
1 (0.2 m/0.5°)	0.77	1.53	2.55	**1.62**	0.47	0.19	1.89	**0.85**
2 (0.5 m/1°)	0.94	1.77	2.83	**1.85**	0.68	0.24	2.74	**1.22**
3 (1 m/2°)	0.92	1.63	2.48	**1.68**	0.70	0.24	2.96	**1.30**
4 (2 m/5°)	1.35	2.31	3.24	**2.30**	0.98	0.43	3.84	**1.75**
5 (4 m/10°)	2.43	4.16	5.57	**4.05**	1.73	1.17	5.15	**2.68**
1 (0.2 m/0.5°)	0.50	1.04	1.88	**1.14**(+70.4%/-29.5%)	0.13	0.07	0.50	**0.23**(+110.1%/-72.5%)
2 (0.5 m/1°)	0.57	1.10	1.91	**1.19**(+72.1%/-35.4%)	0.15	0.08	0.64	**0.29**(+100.0%/-76.2%)
3 (1 m/2°)	0.77	1.37	1.62	**1.25**(+57.5%/-25.2%)	0.23	0.07	0.65	**0.32**(+97.4%/-75.6%)
4 (2 m/5°)	1.33	2.25	3.16	**2.25**(+3.9%/-2.3%)	0.70	0.23	3.37	**1.43**(+21.7%/-18.1%)
5 (4 m/10°)	2.44	4.17	5.58	**4.06**(-0.3%/+0.2%)	1.36	0.83	2.10	**1.43**(+52.4%/-46.7%)

Table 13: **Concurrent localization and visual perturbations across models at severity 3.** CroTTA recovers 17.9% of L2 degradation and 36.5% of collision-rate degradation on average over the five models with positive gaps.

Method	L2(m)↓ 1s	2s	3s	AVG	Col(%)↓ 1s	2s	3s	AVG
ST-P3 (Hu et al., 2022)	2.00	3.90	5.93	**3.94**	0.79	3.17	5.27	**3.08**
UniAD (Hu et al., 2023)	0.92	1.63	2.48	**1.68**	0.70	0.24	2.96	**1.30**
VAD (Jiang et al., 2023)	0.49	0.82	1.21	**0.84**	0.43	0.56	0.94	**0.64**
DistillDrive (Yu et al., 2025)	0.62	1.11	1.70	**1.14**	0.20	0.68	1.42	**0.77**
SparseDrive (Sun et al., 2025)	0.69	1.23	1.87	**1.26**	0.19	0.63	1.45	**0.76**
DiffusionDrive (Liao et al., 2025)	0.54	0.98	1.49	**1.00**	0.00	0.24	0.55	**0.26**
ST-P3 (Hu et al., 2022)	1.93	3.78	5.75	**3.82**(+18.6%/-3.1%)	0.60	2.39	4.07	**2.35**(+49.3%/-23.5%)
UniAD (Hu et al., 2023)	0.77	1.37	1.62	**1.25**(+57.5%/-25.2%)	0.23	0.07	0.65	**0.32**(+97.4%/-75.6%)
VAD (Jiang et al., 2023)	0.46	0.78	1.16	**0.80**(–/-4.8%)	0.34	0.51	0.86	**0.57**(–/-11.4%)
DistillDrive (Yu et al., 2025)	0.61	1.10	1.68	**1.13**(+2.4%/-1.2%)	0.15	0.61	1.35	**0.70**(+9.1%/-8.3%)
SparseDrive (Sun et al., 2025)	0.66	1.16	1.80	**1.21**(+8.9%/-4.5%)	0.00	0.58	1.26	**0.61**(+21.2%/-18.9%)
DiffusionDrive (Liao et al., 2025)	0.53	0.97	1.48	**0.99**(+2.3%/-1.0%)	0.00	0.24	0.52	**0.25**(+5.5%/-3.8%)

collision rate over the five models with positive corruption-induced gaps. Relative reductions across all six models average 6.6% and 23.6%, respectively. The improvements across planners indicate that CroTTA's benefit under joint disturbance extends beyond UniAD and does not require exact localization.

C Generalization under Out-of-Distribution Conditions

We evaluate CroTTA under cross-city distribution shift to assess its generalization to more varied driving conditions. We further examine whether adaptation remains effective in complex scenes with strong interactions between traffic participants.

C.1 Cross-City Transfer

We train SparseDrive exclusively on Boston data, which accounts for 55.8% of the nuScenes training set, using 100 epochs for Stage 1 and 10 epochs for Stage 2. We then evaluate the model in Holland Village, One-North, and Queenstown in Singapore without synthetic corruption. Boston serves as the source-domain reference in Table 14.

CroTTA improves both metrics in all three target towns, recovering 9.9%–35.6% of the L2 gap and 8.0%–45.5% of the collision-rate gap relative to Boston. These improvements show that the adaptation objective also benefits planning under geographic distribution shift, without requiring synthetic corruption to induce the shift.

Table 14: **Cross-city transfer of Boston-trained SparseDrive.** CroTTA improves both metrics across three Singapore towns, recovering 9.9%–35.6% of the L2 gap and 8.0%–45.5% of the collision-rate gap relative to Boston. No synthetic corruption is applied.

Metric	Boston	Holland Village	One-North	Queenstown
L2 (m)↓	0.61	1.32	1.06	1.08
Col. (%)↓	0.01	0.51	0.45	0.10
L2 (m)↓	—	**1.25**(+9.9%/-5.3%)	**0.90**(+35.6%/-15.1%)	**0.98**(+21.3%/-9.3%)
Col. (%)↓	—	**0.47**(+8.0%/-7.8%)	**0.25**(+45.5%/-44.4%)	**0.06**(+44.4%/-40.0%)

C.2 Strong-Interaction Scenarios

To examine adaptation when interactions between traffic participants place greater demands on planning, we select strong-interaction scenes from the Singapore test set following the scenario-mining protocol of Jiang et al. (2025). This subset comprises less than 5% of the Singapore test data. We evaluate the same Boston-trained SparseDrive model on clean inputs and four visual corruptions.

Table 15 shows improvements in both metrics on clean inputs and under each corruption. Relative to the full-validation clean reference in Table 7, the four-corruption average yields 24.5% L2 and 55.4% collision recovery; the relative reductions are 7.3% and 49.8%, respectively. The substantially larger relative reduction in collision rate indicates that adaptation benefits collision avoidance as well as trajectory accuracy in these demanding scenes.

Table 15: **Planning performance on strong-interaction scenes from Singapore.** CroTTA improves both metrics on clean inputs and under all four visual corruptions. AVG excludes Clean; recovery uses the full-validation reference in Table 7.

Metric	Clean	Gaussian	Fog	Snow	Brightness	AVG
L2 (m)↓	0.84	0.98	0.86	0.93	0.81	**0.90**
Col. (%)↓	1.15	1.09	0.51	1.40	0.19	**0.80**
L2 (m)↓	**0.78**(-7.1%)	**0.95**(+8.6%/-3.1%)	**0.74**(+52.2%/-14.0%)	**0.86**(+23.3%/-7.5%)	**0.77**(+22.2%/-4.9%)	**0.83**(+24.5%/-7.3%)
Col. (%)↓	**0.29**(-74.8%)	**0.44**(+64.4%/-59.6%)	**0.09**(+97.7%/-82.4%)	**0.89**(+38.6%/-36.4%)	**0.18**(+9.1%/-5.3%)	**0.40**(+55.4%/-49.8%)

D Sensitivity Analysis of Hyperparameters

We examine sensitivity to the loss weight and learning rate to assess whether CroTTA's gains persist across different adaptation settings.

D.1 Perception–Planning Loss Weight

We vary λ in Eq. 4 to examine the sensitivity to the balance between perception entropy and temporal planning consistency. On UniAD under severe Gaussian noise, we test $\lambda \in \{0.1, 0.3, 1, 3, 10\}$ while keeping the remaining configuration fixed. Table 16 shows that all tested weights improve both metrics over the unadapted model. The default $\lambda = 1$ achieves the lowest L2 error and a tied lowest collision rate at the reported precision. Increasing λ to 10 weakens these gains, indicating that placing too much weight on temporal consistency can reduce the benefit of adaptation.

Table 16: **Sensitivity to the perception–planning loss weight on UniAD.** All tested weights improve both metrics; $\lambda = 1$ achieves the lowest L2 error and a tied lowest collision rate. Bold entries denote the best reported results.

Metric	Without TTA	$\lambda = 0.1$	$\lambda = 0.3$	$\lambda = 1$	$\lambda = 3$	$\lambda = 10$
L2 (m) ↓	1.38	1.28	1.27	**1.25**	1.27	1.32
Col. (%) ↓	0.95	**0.72**	**0.72**	**0.72**	0.82	0.93

We further evaluate the loss weight across all six architectures to assess whether a shared default is suitable beyond UniAD. Table 17 shows that $\lambda = 1$ is optimal in eight of the twelve model–metric combinations, with the remaining optima at 0.3 or 3. These results support its use as a shared default across architectures.

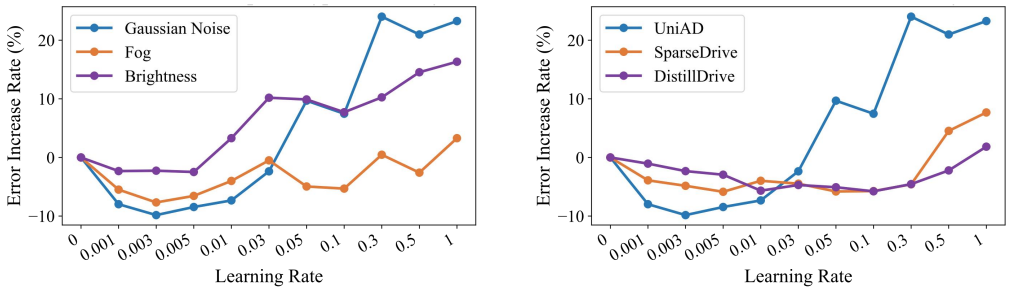
Table 17: **Best loss weight across architectures.** The shared default $\lambda = 1$ is optimal in eight of twelve model–metric combinations; the remaining optima are 0.3 or 3. Columns correspond to ST-P3 (Hu et al., 2022), UniAD (Hu et al., 2023), VAD (Jiang et al., 2023), DistillDrive (Yu et al., 2025), SparseDrive (Sun et al., 2025), and DiffusionDrive (Liao et al., 2025).

Metric	ST-P3	UniAD	VAD	DistillDrive	SparseDrive	DiffusionDrive
Best λ for L2	1	1	0.3	3	3	1
Best λ for Col.	1	1	1	3	1	1

D.2 Learning Rate

The learning rate determines how strongly each observation changes the adapted model. We vary it across corruption types and model architectures to examine the range over which updates improve planning. Figure 3 reports the relative change in error, with negative values indicating improvement over the unadapted model.

The range of beneficial learning rates depends on both the corruption and the architecture. Under Gaussian noise, moderate updates reduce error, whereas larger updates can reverse the gains. Fog is less sensitive across the tested range, while brightness begins to deteriorate at smaller learning rates. Across architectures, SparseDrive and DistillDrive retain improvements over a wider range than UniAD. Thus, although several configurations tolerate a range of update strengths, effective adaptation still depends on choosing a learning rate suited to the model and input conditions.



(a) Corruption Type Sensitivity **(b) Model Sensitivity**

Figure 3: **Learning-rate sensitivity across corruption types and architectures.** Negative error changes indicate improvement over the unadapted model. The range of beneficial learning rates varies with both the corruption and the architecture; large updates can reverse the gains.

E Extended Component Ablation Studies

We extend the component ablations to examine whether the contributions observed on UniAD carry over to other planning architectures. The full UniAD results complement the averages in Table 5, while ST-P3 and SparseDrive provide comparisons using dense and sparse representations. All three experiments use severe Gaussian noise and the same component combinations.

E.1 Full UniAD Results

Table 18 reports the UniAD ablation at each prediction horizon. Combining the perception and temporal objectives increases collision recovery from 4.5% with either objective alone to 21.2%. The full method achieves the highest recovery in both average metrics, at 29.5% for L2 and 34.8%

for collision rate. These results support combining the two adaptation signals with the stabilization mechanism of the full framework.

Table 18: **Ablation study of key components on UniAD.** Joint adaptation yields 21.2% collision recovery; the full method raises this to 34.8% and achieves 29.5% L2 recovery. Component definitions follow Table 5; shading marks the full method.

Component			L2(m)↓				Col(%)↓			
$\mathcal{L}_{mm}$	$\mathcal{L}_{temp}$	$\mathcal{P}_{\Omega}$	1s	2s	3s	AVG	1s	2s	3s	AVG
-	-	-	0.60	1.29	2.25	**1.38**	0.81	0.83	1.23	**0.95**
✓	-	-	0.52	1.24	2.22	**1.33**(+11.4%/-3.6%)	0.70	0.91	1.15	**0.92**(+4.5%/-3.2%)
-	✓	-	0.56	1.23	2.17	**1.32**(+13.6%/-4.3%)	0.81	0.80	1.15	**0.92**(+4.5%/-3.2%)
✓	✓	-	0.52	1.25	2.19	**1.32**(+13.6%/-4.3%)	0.43	0.85	1.16	**0.81**(+21.2%/-14.7%)
✓	✓	✓	0.51	1.15	2.09	**1.25**(+29.5%/-9.4%)	0.53	0.57	1.08	**0.72**(+34.8%/-24.2%)

E.2 Ablation on ST-P3

ST-P3 tests the same components in a dense BEV architecture. In Table 19, combining the objectives improves both average metrics over either objective alone. The full method further raises L2 recovery from 27.9% to 34.3%, while retaining 31.3% collision recovery. The benefit of coordinating adaptation signals therefore extends to a planner with substantially larger errors under corruption.

Table 19: **Ablation study of key components on ST-P3.** Joint adaptation improves recovery in both metrics over either objective alone. The full method achieves the highest L2 recovery of 34.3%, together with 31.3% collision recovery. Notation follows Table 18.

Component			L2(m)↓				Col(%)↓			
$\mathcal{L}_{mm}$	$\mathcal{L}_{temp}$	$\mathcal{P}_{\Omega}$	1s	2s	3s	AVG	1s	2s	3s	AVG
-	-	-	2.51	5.22	8.13	**5.29**	0.89	5.08	9.42	**5.13**
✓	-	-	2.39	4.74	7.16	**4.76**(+26.4%/-10.0%)	0.66	4.13	7.66	**4.15**(+27.8%/-19.1%)
-	✓	-	2.39	4.74	7.19	**4.77**(+25.9%/-9.8%)	0.71	3.96	7.93	**4.20**(+26.4%/-18.1%)
✓	✓	-	2.38	4.70	7.12	**4.73**(+27.9%/-10.6%)	0.66	3.86	7.49	**4.00**(+32.1%/-22.0%)
✓	✓	✓	2.29	4.57	6.95	**4.60**(+34.3%/-13.0%)	0.46	3.88	7.76	**4.03**(+31.3%/-21.4%)

E.3 Ablation on SparseDrive

SparseDrive provides a complementary test using sparse instance queries. Table 20 shows that the full method improves both average metrics over the joint-loss variant, achieving the highest L2 recovery of 26.7% and a tied highest collision recovery of 24.2%. Across the three architectures, the full framework achieves the strongest L2 recovery, while the gains from individual components vary with the planner.

Table 20: **Component ablation on SparseDrive.** The full method achieves 26.7% L2 recovery and 24.2% collision recovery, improving both metrics over the joint-loss variant. Notation follows Table 18.

Component			L2(m)↓				Col(%)↓			
$\mathcal{L}_{mm}$	$\mathcal{L}_{temp}$	$\mathcal{P}_{\Omega}$	1s	2s	3s	AVG	1s	2s	3s	AVG
-	-	-	0.36	0.73	1.25	**0.78**	0.06	0.27	0.90	**0.41**
✓	-	-	0.36	0.71	1.20	**0.75**(+20.0%/-3.8%)	0.06	0.22	0.77	**0.35**(+18.2%/-14.6%)
-	✓	-	0.36	0.71	1.19	**0.75**(+20.0%/-3.8%)	0.13	0.22	0.64	**0.33**(+24.2%/-19.5%)
✓	✓	-	0.35	0.71	1.20	**0.75**(+20.0%/-3.8%)	0.06	0.23	0.76	**0.35**(+18.2%/-14.6%)
✓	✓	✓	0.35	0.69	1.17	**0.74**(+26.7%/-5.1%)	0.12	0.22	0.66	**0.33**(+24.2%/-19.5%)

Table 21: **Effect of extending baseline entropy objectives to the motion head.** Adding motion entropy lowers mean L2 and collision recovery by 4.5 and 10.3 percentage points across five baselines. Unshaded baseline rows use perception entropy; shaded rows also use motion entropy. Both configurations use Without TTA as the corrupted reference.

Method	L2(m)↓				Col(%)↓			
	1s	2s	3s	AVG	1s	2s	3s	AVG
Without TTA	0.60	1.29	2.25	**1.38**	0.81	0.83	1.23	**0.95**
Tent (Wang et al., 2020)	0.58	1.28	2.24	**1.37**(+2.3%/-0.7%)	0.81	0.96	1.05	**0.94**(+1.5%/-1.1%)
EcoTTA (Song et al., 2023)	0.60	1.29	2.24	**1.38**(0.0%/0.0%)	0.83	0.83	1.13	**0.93**(+3.0%/-2.1%)
EATA (Niu et al., 2022)	0.55	1.26	2.23	**1.35**(+6.8%/-2.2%)	0.76	0.75	1.01	**0.84**(+16.7%/-11.6%)
EATA–E (Niu et al., 2022)	0.53	1.25	2.23	**1.34**(+9.1%/-2.9%)	0.71	0.91	1.05	**0.89**(+9.1%/-6.3%)
SAR (Niu et al., 2023)	0.56	1.27	2.24	**1.36**(+4.5%/-1.4%)	0.76	0.78	1.05	**0.86**(+13.6%/-9.5%)
Tent (Wang et al., 2020)	0.53	1.27	2.29	**1.36**(+4.5%/-1.4%)	0.61	0.86	1.38	**0.95**(0.0%/0.0%)
EcoTTA (Song et al., 2023)	0.61	1.34	2.36	**1.44**(-13.6%/+4.3%)	0.81	1.00	1.49	**1.10**(-22.7%/+15.8%)
EATA (Niu et al., 2022)	0.54	1.28	2.29	**1.37**(+2.3%/-0.7%)	0.75	0.65	1.31	**0.90**(+7.6%/-5.3%)
EATA–E (Niu et al., 2022)	0.53	1.26	2.28	**1.36**(+4.5%/-1.4%)	0.66	0.90	1.26	**0.94**(+1.5%/-1.1%)
SAR (Niu et al., 2023)	0.53	1.28	2.29	**1.37**(+2.3%/-0.7%)	0.56	0.86	1.31	**0.91**(+6.1%/-4.2%)

F TTA Baseline Implementations and Task Coupling

We describe how the TTA baselines are applied to UniAD and examine whether extending their entropy objectives to an additional prediction task improves planning. This comparison assesses whether adding another task objective is sufficient to obtain the benefits of CroTTA's cross-task adaptation.

F.1 Implementation Details of TTA Baselines

The baselines in Table 4 adapt probabilistic perception outputs while keeping the downstream planner frozen. Tent, EATA, EATA–E, and SAR update only the LayerNorm affine parameters (γ, β) in the perception classification branches; EcoTTA updates lightweight modules in the perception pathway. The perception objective is the binary entropy of UniAD's ten-class object scores from six decoder classification branches, averaged over branches, queries, and classes.

Tent (Wang et al., 2020) minimizes this entropy directly. EATA (Niu et al., 2022) filters unreliable and redundant frames and uses Fisher-weighted elastic weight consolidation (EWC); EATA–E removes that regularization. SAR (Niu et al., 2023) combines reliability filtering with sharpness-aware updates and model recovery. EcoTTA (Song et al., 2023) minimizes perception entropy through its adaptation modules. These implementations retain the respective adaptation mechanisms while applying them to the same type of probabilistic output.

F.2 Extending Entropy Minimization to Motion Prediction

Motion prediction provides an additional probabilistic output to which entropy minimization can be applied. We therefore extend each baseline's objective to the motion head and compare it with perception-only adaptation in Table 21. The extension lowers collision recovery for all five methods and reduces mean L2 and collision recovery by 4.5 and 10.3 percentage points, respectively. The main comparison uses the better-performing perception-only configurations.

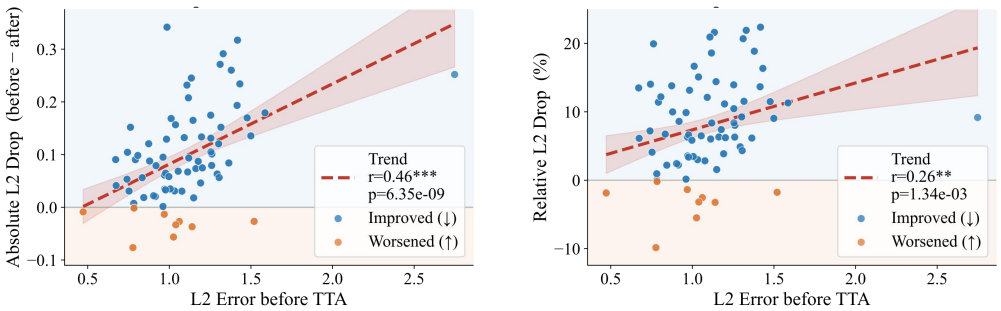
These results show that adding another confidence objective is insufficient to improve planning in this setting. Motion entropy encourages confidence in a predicted mode, whereas CroTTA constrains ego planning through agreement between a previous prediction and subsequent observed displacement. The comparison highlights the importance of the adaptation signal used to connect perception and planning.

G Case Study and Error Correlation Analysis

We complement the aggregate metrics with scenario-level analysis and trajectory visualizations to examine where adaptation helps and where it remains insufficient.

G.1 Scenario-Level Error Analysis

Figure 4 plots absolute and relative L2 reductions against the unadapted error under Gaussian noise at severity 5. Scenarios with larger initial errors tend to show larger reductions, although adaptation also increases error in some cases. The trend suggests that CroTTA can correct substantial initial deviations, while the variation motivates examining both successful corrections and remaining failures.



(a) Absolute L2 Improvement **(b) Relative L2 Improvement**

Figure 4: **Scenario-level L2 changes under severe Gaussian noise.** Absolute (a) and relative (b) reductions are plotted against the unadapted L2 error. Blue and orange points denote improved and degraded scenarios, respectively; fitted trends and correlation statistics summarize the relationship with initial error.

G.2 Qualitative Success and Failure Cases

Figure 5 compares trajectory corrections with cases where adaptation does not reduce L2 error, illustrating the variation observed in the scenario-level analysis.

Successful corrections (Cases 1 and 2). Under Gaussian noise and severe brightness corruption, respectively, the adapted trajectories align more closely with the ground truth in straight-ahead and turning scenarios. These examples illustrate how adaptation corrects visible planning deviations under different visual disturbances.

Remaining failures (Cases 3 and 4). In Case 3, evaluated with a relatively high learning rate, adaptation moves the prediction to the opposite side of the ground truth. This behavior is consistent with an overly large update and complements the sensitivity analysis in Appendix D. In Case 4, the adapted trajectory appears smoother but has higher L2 error than the oscillatory baseline, illustrating that pointwise displacement accuracy and qualitative trajectory smoothness can disagree.

G.3 Adaptation across Consecutive Frames

To visualize how updates affect successive predictions, Figure 6 follows frames 7, 10, and 15 of a sequence with brightness corruption. The adapted prediction moves closer to the ground-truth trajectory over the displayed frames, illustrating correction as new observations become available.

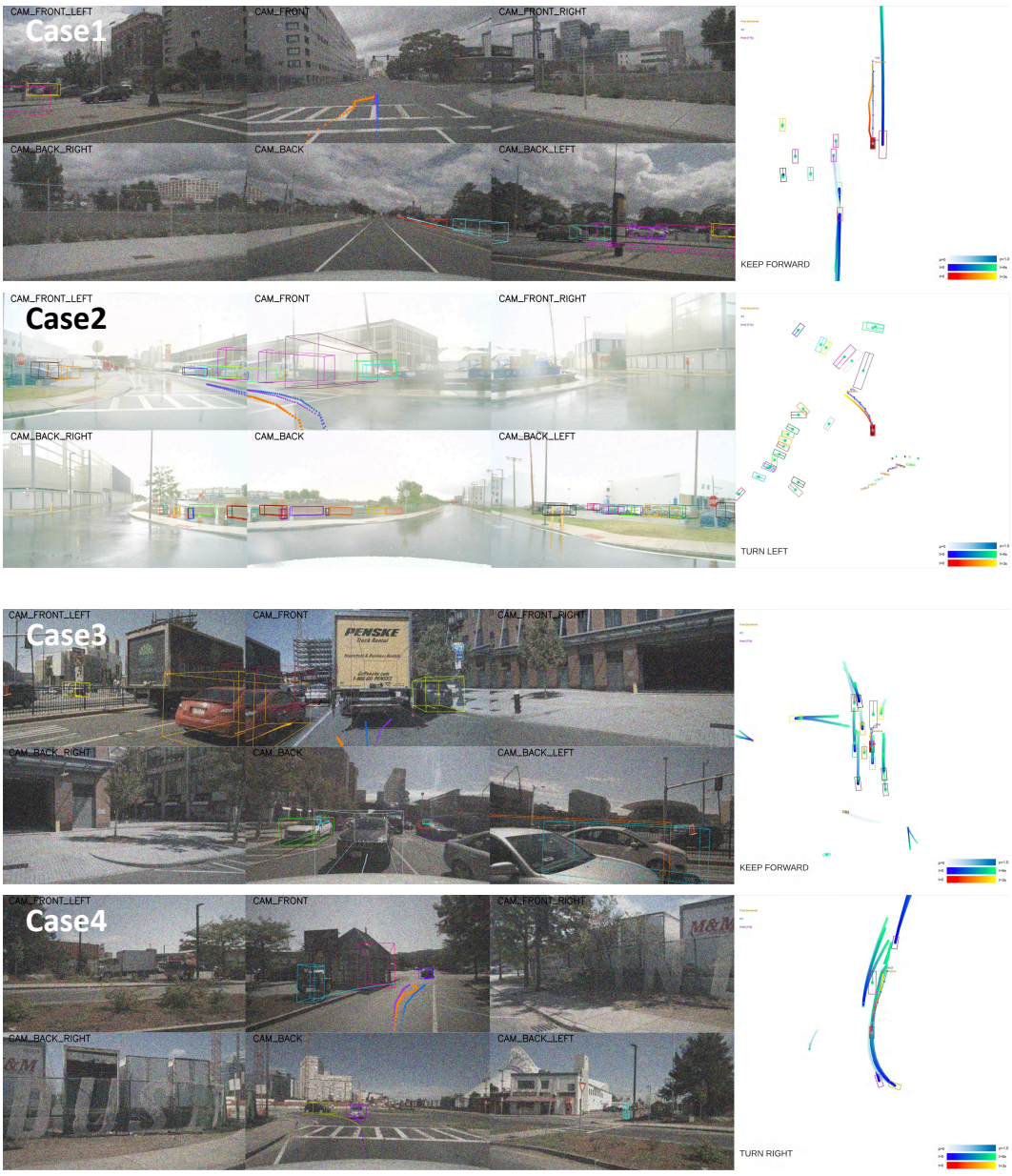
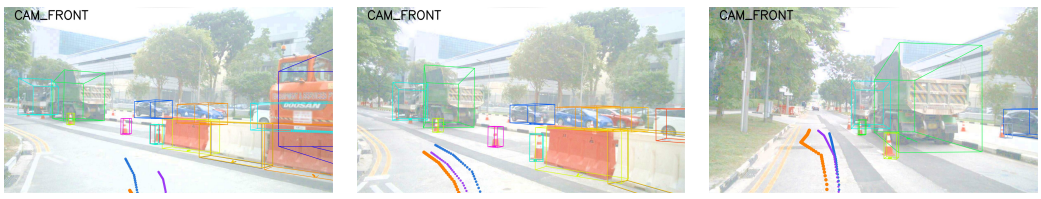


Figure 5: **Qualitative trajectory comparisons before and after TTA.** Blue, orange, and purple lines denote the ground truth, unadapted prediction, and CroTTA prediction, respectively. Cases 1–2 illustrate successful corrections; Cases 3–4 show remaining failures.



(a) frame=7 **(b) frame=10** **(c) frame=15**

Figure 6: **Trajectory adaptation across frames under brightness corruption.** At frames 7, 10, and 15, the purple adapted trajectory progressively approaches the blue ground truth; orange denotes the unadapted prediction.

H LIMITATIONS AND FUTURE WORK

By enabling end-to-end autonomous driving models to adapt dynamically to unpredictable environmental shifts, our framework enhances the overall robustness and safety. This progress facilitates the more reliable deployment of autonomous vehicles in complex, real-world conditions where current models typically struggle. In this section, we discuss the inherent limitations of our current framework based on the empirical observations from Appendix C and the corresponding experiments in Appendix A, followed by potential directions for future research.

Limitations. Through a comparative analysis of different corruption patterns and architectures, we identify two primary limitations: (i) *Efficacy Gap Between Various Distribution Shifts:* We observe that the effectiveness of CroTTA is highly dependent on the nature of the distribution shift. While the model can recover a significant portion of performance in structured environments (e.g., Fog or Brightness), its efficacy diminishes when encountering destructive corruptions like Gaussian noise. This suggests that when a shift effectively washes out essential spatial features, there is a potential risk in how much performance can be clawed back through test-time alone. (ii) *Dependency on Hyperparameter Stability:* Our sensitivity analysis reveals that while modern architectures offer a wide optimization buffer, earlier frameworks like UniAD exhibit higher sensitivity to the learning rate. In real-world, real-time autonomous driving scenarios, the system operates blindly without ground truth or the luxury of "trial-and-error" to identify the optimal learning rate. Consequently, an improper parameter setting could potentially lead to performance collapse during online adaptation.

Future Work. To address these limitations, our future research will focus on the following directions: (i) *Self-tuning Adaptation Mechanisms:* Instead of relying on a fixed learning rate, future work could explore mechanisms that automatically modulate the adaptation strength based on the loss and gradient. Developing a system that can autonomously determine "when to adapt" and "how much to update" would significantly enhance the plug-and-play capability of TTA frameworks. (ii) *Safety-constrained Online Optimization:* Given the high stakes of autonomous driving, incorporating explicit safety constraints into the adaptation objective is a vital direction. By ensuring that parameter updates are always bounded by a "safety envelope," we can prevent hazardous planning decisions even during rapid environmental transitions or under extreme noise conditions.