CoFiDrive: Scene-Conditioned Registers and Coarse-to-Fine Trajectory Scoring for End-to-End Driving

Anonymous authors
Paper under double-blind review

Abstract

Vision-based driving planners must compress dense multi-camera observations while retaining the information needed to generate and compare motion alternatives. Road layout and traffic configuration provide scene context, whereas nearby trajectories can differ in their clearance from vehicles or lane boundaries. We present CoFiDrive, which combines scene-conditioned register refinement with coarse-to-fine trajectory scoring. Four camera views contribute 16 learned registers each, forming a compact 64-token scene memory. A semantic token compressor (TC) refines these 64 tokens through scene-conditioned soft expert mixing, preserving the memory size for trajectory generation and scoring. The Cross-Compression Scorer (CCS) combines the refined scene features with spatially pooled patch features through candidate-dependent gated attention, giving each proposal access to coarse context and local spatial detail. During training, control-space augmentation (CSA) derives acceleration and yaw-rate sequences from PDM-ranked proposals, perturbs these controls, and uses bicycle-model rollouts to create additional candidates. The PDM evaluator supplies component-score targets for the same CCS on the original and augmented candidates. CSA broadens CCS supervision without adding model parameters. At inference, CCS ranks only the original 64 proposals, with no augmentation or PDM target computation. After 10 training epochs, CoFiDrive achieves 92.4 and 92.9 PDMS on NAVSIM-v1 using train85k and trainval103k, respectively, and 48.0 EPDMS on NAVSIM-v2 NavHard.

1 Introduction

Vision-based driving planners must generate plausible trajectories and select one for execution. Two candidates can follow the same road yet differ in vehicle clearance, distance to a lane boundary, or progress. Scene context informs both generation and evaluation, while spatial details help distinguish similar motions. A compact visual representation must therefore retain useful information about the scene and support comparison of the resulting candidates.

How can a planner refine compact scene features for trajectory generation while preserving access to the spatial evidence needed to compare its candidates?

End-to-end driving systems encode observations into learned scene features for planning (Chitta et al., 2023; Hu et al., 2023). Proposal-based planners make motion alternatives explicit and rank them with learned scores (Li et al., 2024; Guo et al., 2026; Liao et al., 2025). Multi-camera images produce long patch sequences with repeated background content. Latent representations and token selection reduce the visual sequence processed downstream (Jaegle et al., 2021; Ryoo et al., 2021; Rao et al., 2021). For driving, compactness leaves two related design choices: how to adapt the retained features to changing road layouts and traffic, and how to combine them with spatial evidence when evaluating trajectories.

We address these choices with CoFiDrive. Its encoder follows the camera-register bottleneck of DrivoR (Kirby et al., 2026): four views with 16 learned registers each produce a 64-token scene memory from 15,744 image-patch states. The semantic token compressor (TC) adapts the register features to the current scene through soft mixing over six SwiGLU experts. Pooled scene content

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.

determines the mixture weights shared by all tokens, and every expert remains active. The refined memory conditions both trajectory generation and scoring. The inherited register bottleneck supplies the token-count reduction, while TC changes the retained features and preserves the 64-token interface.

The Cross-Compression Scorer (CCS) complements the scene memory with spatially pooled patch features from all four views. Each candidate queries the two memories in parallel. A candidate-dependent gate combines their responses before the metric heads predict trajectory quality. This Dual-Channel design lets scoring use local spatial evidence alongside the refined scene features, while the generator keeps its compact visual input. Here, coarse-to-fine describes the combination of coarse scene context and finer spatial evidence within each scoring block, not successive filtering of the proposal set.

The scoring task also depends on the motion alternatives available during training. Control-space augmentation (CSA) expands this set around promising proposals so that CCS receives supervision on nearby motions with different control sequences. PDM scores rank the 64 generated trajectories, and the top 16 become seeds. CSA perturbs each seed's acceleration and yaw rate, then uses bicycle-model rollouts to fill 12 candidate slots per seed. These 192 entries join the original 64 in an auxiliary scoring call that shares CCS parameters. PDM component scores supply targets for both lists. Section 3 specifies rollout checks and seed fallback. At inference, CSA and PDM target computation are omitted, and CCS ranks the original 64 proposals.

Our contributions are:

- **Scene-conditioned register refinement.** TC uses dense expert mixing to adapt the compact scene features shared by the generator and scorer, preserving the 64-token memory.
- **Cross-compression trajectory scoring.** CCS combines scene context and spatial patch evidence through candidate-dependent gated attention while keeping the generator input compact.
- **Control-space augmentation.** CSA trains CCS on additional motion alternatives using benchmark component scores, with no added parameters or inference candidates.
- **Benchmark results.** With 10-epoch training, CoFiDrive reaches 92.4 and 92.9 PDMS on NAVSIM-v1 for train85k and trainval103k, respectively, and 48.0 EPDMS on NavHard.

2 Related Work

2.1 End-to-End and Proposal-Based Driving

End-to-end driving methods organize visual information around the planning task. TransFuser (Chitta et al., 2023) fuses camera and LiDAR features, while TCP (Wu et al., 2022) couples trajectory and control prediction. UniAD (Hu et al., 2023) connects perception, prediction, and planning through task queries. PARA-Drive (Weng et al., 2024) parallelizes related components, and VADv2 (Jiang et al., 2026) combines vectorized scene representations with probabilistic planning. SparseDrive (Sun et al., 2025) organizes the stack around a sparse scene representation. Our focus is the visual bottleneck: adapting compact scene features while retaining spatial evidence for candidate evaluation.

Proposal-based planners represent alternative behaviors explicitly. Hydra-MDP and Hydra-MDP++ train metric-prediction heads using human and rule-based supervision (Li et al., 2024; 2025). iPad (Guo et al., 2026) refines proposals iteratively, and DiffusionDrive (Liao et al., 2025) generates them through truncated diffusion. DriveSuprim (Yao et al., 2026) progressively filters candidates, whereas CCS combines coarse scene context and finer patch evidence for every proposal. DrivoR (Kirby et al., 2026) couples camera registers with a trajectory decoder and component-wise scorer. CoFiDrive retains this generator and inference candidate count. It adds TC for scene-conditioned register refinement and CCS for scoring from scene and patch memories. CSA supplies additional candidates only while training CCS.

Control-space refinement can also modify proposals at test time. TOAD (Xu et al., 2026) maps trajectories to acceleration and yaw-rate controls, perturbs them locally, and optimizes the resulting bicycle-model rollouts against a frozen scorer. CSA uses a related control parameterization to

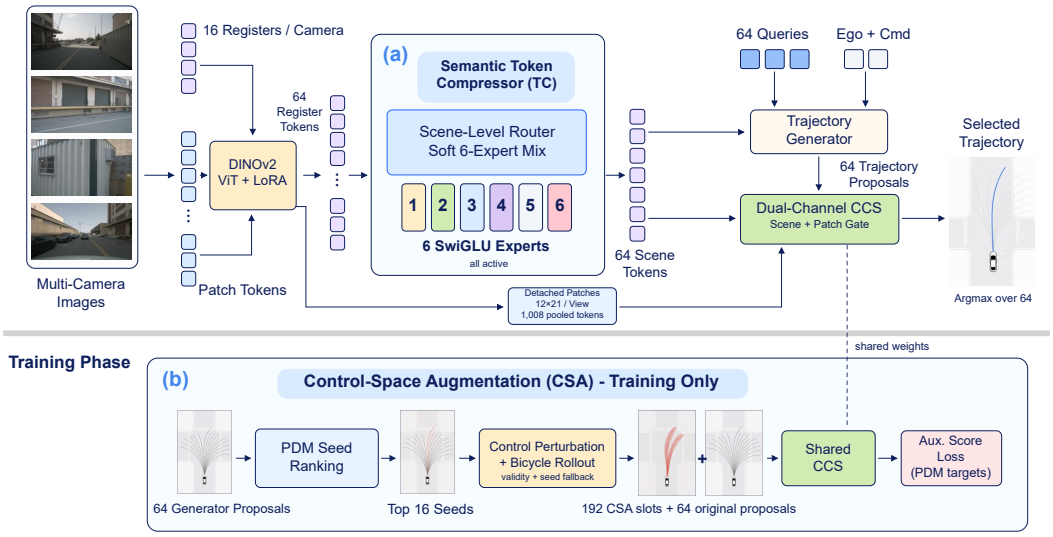

Figure 1: **CoFiDrive architecture.** (a) TC refines the 64 camera-register tokens shared by the generator and CCS; CCS also queries pooled patch features. (b) Training-only CSA allocates 12 control-space slots to each of 16 PDM-ranked seeds. These 192 slots and the original 64 proposals form an auxiliary list; unsuccessful slots retain their seeds. The auxiliary call shares CCS parameters and visual memories with the main call; the dashed link denotes shared parameters. Inference ranks only the original 64 proposals.

produce scorer-training examples; the perturbations and PDM oracle are absent at inference, where CCS ranks the original proposal set.

2.2 Visual Tokens, Registers, and Compression

Vision Transformers (Dosovitskiy et al., 2021) encode images as patch-token sequences. Compact alternatives include the latent array in Perceiver (Jaegle et al., 2021), adaptive token extraction in TokenLearner (Ryoo et al., 2021), and learned token pruning in DynamicViT (Rao et al., 2021). These methods change the length or organization of the visual sequence processed downstream.

Register tokens were introduced to absorb high-norm artifacts in low-information regions of vision transformers (Darcet et al., 2024). DrivoR (Kirby et al., 2026) uses additional per-camera registers as learned compression tokens for planning. CoFiDrive inherits this bottleneck and changes how the retained features are processed and used. TC adapts register features to scene content without further reducing token count. CCS combines the resulting scene memory with spatially organized patch features when scoring trajectories.

2.3 Mixture-of-Experts Models

Sparse Mixture-of-Experts models route tokens through a subset of networks to increase conditional capacity (Shazeer et al., 2017; Fedus et al., 2022). V-MoE (Riquelme et al., 2021) applies this principle to vision-transformer feed-forward layers. Expert-choice routing lets each expert select a fixed-capacity token set (Zhou et al., 2022), while Soft MoE (Puigcerver et al., 2024) forms differentiable mixtures over expert slots. In contrast, TC uses dense mixing after register compression. Every expert processes every retained token, and pooled scene content determines one mixture shared across the scene. Its conditional behavior comes from changing the mixture weights, not from sparse dispatch or reduced expert computation.

3 METHOD

3.1 PLANNING FROM A COMPACT VISUAL MEMORY

At time t, the model receives the latest images $\mathcal{I}_t = \{I_t^c\}_{c=1}^C$ from four cameras, the ego state $s_t \in \mathbb{R}^7$ (pose, velocity, and acceleration), and a driving command $u_t \in \mathbb{R}^4$. A linear map encodes $[s_t; u_t]$ as a 256-dimensional ego token. The generator predicts a list of $N_d = 64$ proposals, $\mathcal{V}_{\text{dyn}} = (\tau_j)_{j=1}^{N_d}$, each containing $T = 8$ future poses (x, y, θ) in the current ego frame. A scorer selects the executed plan:

$$\tau^\star = \tau_{j^\star}, \qquad j^\star = \arg\max_{1 \le j \le N_d} S_j(\mathcal{V}_{\text{dyn}} \mid \mathcal{I}_t, s_t, u_t). \tag{1}$$

Here, S_j is the score assigned to proposal j in the supplied proposal set. Figure 1 shows the three components of CoFiDrive. TC refines the camera-register features; CCS combines scene context with spatial patch evidence; and control-space augmentation (CSA) expands the candidates used to train CCS. We retain the camera-register encoder and query-based generator of DrivoR (Kirby et al., 2026).

Camera registers. For camera c, let $P_0^c \in \mathbb{R}^{M \times d_v}$ denote the patch embeddings with positional encoding, and let q_{cls} and Q_{pre}^c denote the classification token and pretrained backbone registers. We prepend R learned camera registers Q_{reg}^c and encode each view independently with a shared ViT:

$$H^c = f_\theta([Q_{\text{reg}}^c; q_{\text{cls}}; Q_{\text{pre}}^c; P_0^c]). \tag{2}$$

The encoded camera registers $Z^c = H_{\text{reg}}^c$ form the compact path; encoded patch states $P^c = H_{\text{patch}}^c$ are retained separately for scoring. We use DINOv2 ViT-S/14 with $d_v = 384$, frozen pretrained weights, and trainable LoRA adapters. Following DrivoR, $C = 4$ and $R = 16$, giving 64 camera-register outputs. A linear projection maps these outputs to width $d = 256$ before TC.

3.2 SCENE-CONDITIONED REGISTER EXPERTS

A fixed-size register memory must represent different road layouts and traffic configurations. The semantic token compressor (TC) adapts its transformation through an input-dependent expert mixture. It maps the projected registers $X = (x_i)_{i=1}^{CR}$ through $E = 6$ experts. A bias-free linear router maps their mean to the mixture weights:

$$\bar{x} = \frac{1}{CR} \sum_{i=1}^{CR} x_i, \qquad p = \text{softmax}(W_r \bar{x} + \xi). \tag{3}$$

Gaussian logit jitter $\xi \sim \mathcal{N}(0, 0.1^2 I)$ is applied during training and set to zero at inference. Routing uses visual content alone, with the same weights for every token in a scene. All experts process all tokens; there is no top-k dispatch.

Each expert F_e is a SwiGLU feed-forward network with 1,024 hidden units and dropout 0.1 before its output projection. The mixture forms a scaled residual:

$$\hat{z}_i = x_i + \alpha_{\text{moe}} \sum_{e=1}^{E} p_e F_e(x_i), \tag{4}$$

where α_{moe} is learned and initialized to one. The resulting scene memory $\hat{Z} = (\hat{z}_i)_{i=1}^{CR}$ still contains 64 tokens and is shared by the generator and scorer. Token-count reduction occurs in the inherited camera-register bottleneck, not in this expert transformation.

Trajectory generation. We add the ego token to 64 learned trajectory queries. Four decoder blocks apply query self-attention, cross-attention to $\hat{Z}$, and feed-forward refinement. An MLP maps each final query to $\tau_j \in \mathbb{R}^{T \times 3}$. The generator uses best-of-64 regression:

$$\mathcal{L}_{\text{traj}} = \min_{1 \le j \le N_d} d(\tau_j, \tau_{\text{gt}}), \tag{5}$$

where d sums the absolute coordinate errors at each pose and averages across poses. This is the single-target objective. The trainval103k setting adds a separate best-of-64 term for a resampled longer target (Appendix B).

3.3 CROSS-COMPRESSION SCORER (CCS)

CCS queries scene and patch memories in parallel within each scorer block. Following DrivoR, an MLP embeds detached trajectory coordinates as $q_\tau = \psi(\mathrm{stopgrad}(\tau))$. Each candidate attends to:

- **Scene memory:** $\hat{Z}$, augmented with learned camera and token-type embeddings and then layer-normalized.
- **Patch memory:** detached encoded patch states P^c, adaptively averaged to a 12×21 grid per view and projected to width 256. Two-dimensional position, camera, and token-type embeddings are added before layer normalization. The four views yield 1,008 patch tokens.

The patch path bypasses TC. Each of four scorer blocks applies self-attention across the candidate representations, followed by separate cross-attention to the two memories. Consequently, a proposal's score is conditioned on the proposal set supplied to that scorer call, as indicated in Eq. (1). Let h denote a candidate representation after the self-attention residual. The cross-attention increments Δ_s and Δ_p are fused as

$$\Delta = (1-g)\Delta_s + g\Delta_p, \qquad g = \mathrm{sigmoid}(\mathrm{MLP}([h; \Delta_s; \Delta_p])). \tag{6}$$

The scalar gate g is candidate-dependent. Residual addition and a feed-forward block complete the update. After four blocks, the ego token is added and six heads predict PDM components. The configured ranking score aggregates NC, DAC, TTC, EP, and comfort; DDC is supervised but omitted from this inference aggregation. Appendix B gives the exact score and loss definitions.

3.4 CONTROL-SPACE AUGMENTATION (CSA)

The main scoring loss supervises the 64 generated proposals. During training, CSA adds nearby candidates by perturbing their control sequences. PDM scores $G_{\mathrm{PDM}}(\tau_j)$ rank the detached proposals, and the top $K = 16$ become seeds. A kinematic projection maps each seed to longitudinal acceleration and yaw-rate sequences, using the current longitudinal ego speed v_0. Each seed receives $V = 12$ augmentation slots:

$$c_i = \mathrm{BM}^{\dagger}(\mathrm{stopgrad}(\tau_i), v_0), \qquad \tilde{\tau}_{i,r} = \mathrm{BM}(\mathrm{clip}(c_i + \delta c_{i,r}), v_0). \tag{7}$$

Each control sequence receives a trajectory-level Gaussian bias and temporally correlated noise. The perturbed controls are clipped and rolled out with the bicycle model; each rollout is checked for velocity and deviation from its seed. A slot keeps the first valid rollout, with at most four attempts. If all attempts fail, that slot contains the seed. Appendix B gives the noise distribution, bounds, and validity thresholds.

The auxiliary candidate list concatenates the original proposals and the retained slots:

$$\mathcal{V}_{\mathrm{aux}} = \mathrm{concat}(\mathcal{V}_{\mathrm{dyn}}, \mathcal{V}_{\mathrm{aug}}), \qquad |\mathcal{V}_{\mathrm{aux}}| = 64 + 16 \times 12 = 256. \tag{8}$$

Seed fallback can produce duplicates. The original and auxiliary lists are scored in separate calls with shared CCS parameters and the same visual memories. CSA therefore changes the training candidate list, not the scorer architecture. Because proposal self-attention operates over each supplied list, its predictions are list-conditioned. The original proposals contribute to both losses using cached PDM component targets; the retained augmentation entries receive their own PDM targets.

3.5 OBJECTIVE AND GRADIENT PATHS

For $q \in \{\mathrm{main}, \mathrm{aux}\}$, component-wise binary cross-entropy supervises the six predicted scores on $\mathcal{V}_q$, with $\mathcal{V}_{\mathrm{main}} = \mathcal{V}_{\mathrm{dyn}}$. Component-specific target mapping and the TTC validity mask are given in Appendix B. The objective is

$$\begin{aligned}\mathcal{L} = \mathcal{L}_{\mathrm{traj}} + \mathcal{L}^{\mathrm{main}}_{\mathrm{score}} + 0.5\mathcal{L}^{\mathrm{aux}}_{\mathrm{score}} \\ + 0.01\mathcal{L}_{\mathrm{lb}} + 0.001\mathcal{L}_{\mathrm{ent}},\end{aligned} \tag{9}$$

where the last two terms regularize load balance and routing entropy.

Score gradients cannot pass through detached trajectory coordinates. Through scene memory, they update TC, the register projection, camera registers, and LoRA adapters. The patch path detaches at the backbone output: its score gradients update the patch projection, learned embeddings, normalization layers, and scorer, not the backbone. Pretrained ViT weights remain frozen. At inference, CSA and PDM target computation are omitted; CCS remains active and ranks the original 64 proposals.



Figure 2: **Intersection trajectory selection on NAVSIM-v1 navtest.** DrivoR (top) and CoFiDrive (bottom) run on the same scene. Blue curves are generated proposals, yellow marks each planner's selection, and white is ground truth. In this case, the CoFiDrive selection follows the reference path more closely; each planner generates and scores its own proposals.

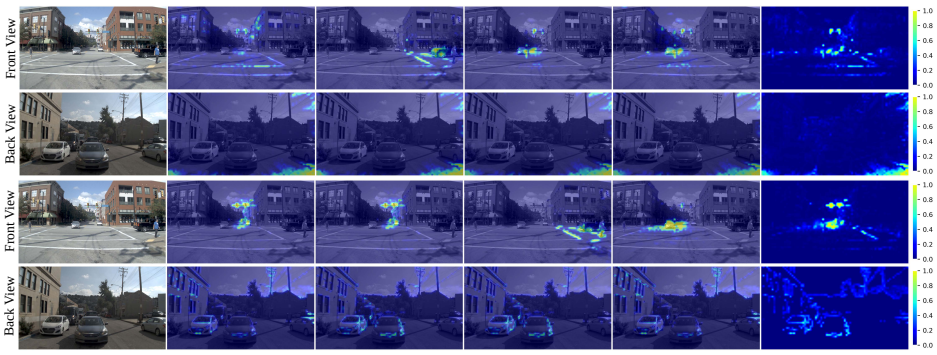

Figure 3: **Register attention in front and rear views.** DrivoR occupies the upper rows and CoFiDrive the lower rows. Each row shows RGB, heatmaps for selected registers 0, 4, 8, and 12, and the accompanying mean map. Warmer colors indicate higher displayed attention. The maps illustrate where the registers attend in these scenes.

4 Experiments

We evaluate CoFiDrive on NAVSIM-v1 and NavHard. The ablations compare sequential additions of TC, Dual-Channel CCS, and CSA training, then test selected routing, fusion, and supervision choices.

4.1 Datasets and Metrics

For NAVSIM-v1 (Dauner et al., 2024), we train separate models on train85k and trainval103k and evaluate on navtest. The evaluation split is not used for training. PDMS combines no at-fault collision (NC), drivable-area compliance (DAC), time to collision (TTC), comfort (Comf.), and ego progress (EP). We also evaluate on NAVSIM-v2 NavHard (Cao et al., 2025) using EPDMS, which adds driving-direction compliance (DDC), traffic-light compliance (TLC), and lane keeping (LK), and replaces the original comfort term with history comfort (HC) and extended comfort (EC). Its second stage evaluates synthetic continuations under ego-pose perturbations. Ablations report NAVSIM-v1 PDMS. Appendix A gives the metric definitions and weights.

4.2 Implementation Details

We train all configurations studied here for 10 epochs, including the ablations and the DrivoR baseline on NavHard. On NavHard, CoFiDrive and DrivoR use the same training split, 10-epoch budget,

and evaluation set. Other methods retain the results reported in the cited publications; these comparisons provide benchmark context, not a matched-compute evaluation.

The visual encoder is DINOv2 ViT-S/14 with LoRA rank 32. We use four cameras, 16 registers per camera, 64 trajectory queries, four decoder layers, four scorer layers, and hidden dimension 256. TC contains six SwiGLU experts, all active under scene-level soft routing. During training, CSA adds 192 slots to the 64 generated proposals for auxiliary scorer supervision. Each slot retains a valid perturbed rollout or, after four failed attempts, its seed. The main scoring loss remains on the original proposals. At inference, augmentation and PDM labels are absent; CCS still scores the original 64. Appendix B reports the optimizer, training budget, loss weights, augmentation settings, and architectural hyperparameters.

Each camera produces 3,936 patch states and 16 register states. The generator thus receives 64 registers instead of 15,744 patch tokens, a 99.6% shorter visual sequence. This inherited bottleneck excludes the scorer's 1,008 pooled patch tokens. We report parameter counts for the added scoring branch, not a measured reduction in end-to-end latency or memory.

4.3 Main Comparison

For the train85k main comparison, CoFiDrive is trained with three independent random seeds for 10 epochs per run and evaluated on the same navtest split. Its reported 92.4 PDMS is the mean across these runs. The trainval103k model reaches 92.9 PDMS; both settings retain 64 scene tokens and 64 inference candidates (Table 1). The 103k configuration also uses a resampled long-trajectory target (Appendix B), so the difference does not isolate training-set size. Relative to train85k, EP rises by 1.3 points and DAC by 0.2, while TTC falls by 0.2 and NC and comfort are unchanged.

For context, DrivoR reports 93.1 and 93.7 PDMS on the NAVSIM-v1 train and trainval settings, respectively (Kirby et al., 2026). These published scores are distinct from our local reproductions. The comparisons below assess the reported benchmark results and the controlled component additions, not a new state-of-the-art ranking.

Table 1: **NAVSIM-v1 navtest results.** CoFiDrive uses 10 epochs on each indicated split; cited baselines retain their reported protocols. UniAD, LTF, and PARA-Drive results follow the NAVSIM evaluation (Dauner et al., 2024). PDMS is on a 0–100 scale.

Method	NC	DAC	TTC	Comf.	EP	PDMS
Ego-stat. MLP (Dauner et al., 2024)	93.0	77.3	83.6	100.0	62.8	65.6
UniVLA (base) (Wang et al., 2026)	96.9	91.1	91.7	96.7	76.8	81.7
UniAD (Hu et al., 2023)	97.8	91.9	92.9	100.0	78.8	83.4
LTF (Chitta et al., 2023)	97.4	92.8	92.4	100.0	79.0	83.8
PARA-Drive (Weng et al., 2024)	97.9	92.4	93.0	99.8	79.3	84.0
DriveX-S (Shi et al., 2025)	97.5	94.0	93.0	100.0	79.7	84.5
World4Drive (Zheng et al., 2025)	97.4	94.3	92.8	100.0	79.9	85.1
DRAMA (Yuan et al., 2024)	98.0	93.1	94.8	100.0	80.1	85.5
PRIX (Wozniak et al., 2026)	98.1	96.3	94.1	100.0	82.3	87.8
DiffusionDrive (Liao et al., 2025)	98.2	96.2	94.7	100.0	82.2	88.1
DIVER (Song et al., 2026)	98.5	96.5	94.9	100.0	82.6	88.3
AutoVLA (Post-RFT) (Zhou et al., 2025)	98.4	95.6	98.0	99.9	81.9	89.1
VADv2 (Jiang et al., 2026)	98.3	97.4	95.7	100.0	82.3	89.3
DriveVLA-W0 (query) (Li et al., 2026a)	98.7	99.1	95.3	99.3	83.3	90.2
ReCogDrive (Li et al., 2026b)	97.9	97.3	94.9	100.0	87.3	90.8
Hydra-MDP++ (V2-99) (Li et al., 2025)	98.6	98.6	95.1	100.0	85.7	91.0
iPad (Guo et al., 2026)	98.6	98.3	94.9	100.0	88.0	91.7
CoFiDrive (train85k)	99.0	98.2	96.2	100.0	88.3	**92.4**
CoFiDrive (trainval103k)	99.0	98.4	96.0	100.0	89.6	**92.9**

On NAVSIM-v2 NavHard, CoFiDrive reaches 48.0 EPDMS versus 47.3 for the DrivoR model reproduced under the same 10-epoch training and evaluation protocol. DrivoR's published NavHard score is 48.3 EPDMS with 10-epoch training (Kirby et al., 2026); the 0.7-point gain here is over the local 47.3 reproduction, not that published result. Equal training duration alone does not establish equivalence with the published configuration. The component changes are mixed: Stage 2 DAC and

Table 2: **NAVSIM-v2 NavHard results.** Stage 1 uses original observations and Stage 2 synthetic continuations. RAP-DINO (Feng et al., 2026) and GTRS-DrivoR (Kirby et al., 2026) use the metric-corrected results reported in DrivoR. CoFiDrive and the local DrivoR baseline use the same 10-epoch protocol. The 47.3 DrivoR row is our reproduction, not its published score. EPDMS is on a 0–100 scale.

Method	Stage 1									Stage 2									EPDMS
	NC	DAC	DDC	TLC	EP	TTC	LK	HC	EC	NC	DAC	DDC	TLC	EP	TTC	LK	HC	EC	
RAP-DINO (Feng et al., 2026)	97.1	94.4	98.8	99.8	83.9	96.9	94.7	96.4	66.2	83.2	83.9	87.4	98.0	86.9	80.4	52.3	95.2	52.4	39.6
GTRS-DrivoR (Kirby et al., 2026)	98.0	95.8	99.7	99.3	72.9	98.2	95.6	96.9	51.6	91.6	86.7	90.2	98.8	73.2	88.9	51.9	94.9	46.4	45.8
DrivoR (Kirby et al., 2026)	98.9	95.3	99.1	99.8	73.0	99.1	93.8	97.6	64.0	91.6	87.3	92.3	98.2	68.6	89.2	50.8	98.5	76.8	47.3
CoFiDrive	99.1	95.3	99.0	99.6	73.9	98.7	93.6	97.6	69.3	89.6	89.3	91.8	98.7	69.7	87.6	53.3	98.7	73.9	**48.0**

LK rise by 2.0 and 2.5 points, while NC, TTC, and EC fall by 2.0, 1.6, and 2.9 points. The aggregate gain should therefore not be read as a uniform improvement across safety and compliance measures. The configured inference score also omits DDC (Appendix B), so the EPDMS gain does not show that the ranking score directly optimizes every reported NavHard component.

4.4 Component Ablation

Table 3 adds TC, Dual-Channel CCS, and CSA in sequence, with the camera encoder, decoder depth, train85k split, and 64-candidate inference protocol fixed. The all-No row is a DrivoR-style base: a camera-register encoder, query-based generator, and scene-memory scorer without TC, the patch branch, or CSA. Its 91.70 is a local ablation result, not the published DrivoR score. Table 4 examines token and expert counts. Every ablation uses 10 epochs. Tables B.4 and B.5 in Appendix B compare register processing, fusion, and repeated candidate supervision.

Table 3: **Sequential component ablation on NAVSIM-v1 navtest.** Train85k, 10 epochs, and 64 inference candidates are fixed. Counts are in M and PDMS is on a 0–100 scale; GTRS and ZTRS parameter counts are author-measured and provide architecture context, not score baselines.

Method	TC	Dual-Channel CCS	CSA	Total parameters	Scorer parameters	PDMS ↑
GTRS V2-99 (Kirby et al., 2026)	–	–	–	82.95M	9.46M	–
ZTRS V2-99 (Li et al., 2026c)	–	–	–	83.38M	9.89M	–
CoFiDrive	No	No	No	40.76M	5.08M	91.70
	Yes	No	No	45.51M	5.80M	91.80
	Yes	Yes	No	47.52M	7.81M	91.90
	Yes	Yes	Yes	47.52M	7.81M	**92.40**

Table 4: **Token and expert sensitivity.** Separate 10-epoch NAVSIM-v1 sweeps train on train85k and evaluate on navtest, with Dual-Channel CCS and CSA enabled. Token counts are per camera; all experts are active. PDMS is on a 0–100 scale.

Tokens/camera	16	32	64	Experts	4	6	8
PDMS ↑	**91.85**	91.66	91.66	PDMS ↑	91.54	**91.85**	91.59

Sequential component ablation. Adding TC alone raises PDMS from 91.70 to 91.80 (Table 3). Dual-Channel CCS then increases it to 91.90, and CSA reaches 92.40. The successive gains are 0.10, 0.10, and 0.50 points, totaling 0.70 over the base configuration in the reported runs. TC improves the point estimate before either patch memory or CSA is introduced; CSA provides the largest observed gain in this sequence. Each gain is conditional on the preceding components, rather than an independent main effect.

Routing, fusion, and supervision controls. With Dual-Channel CCS and CSA fixed, scene routing reaches 92.40 PDMS, versus 92.18 for uniform expert mixing and 92.12 for a single-SwiGLU

control (Table B.4). With both memories fixed, candidate-dependent gating scores 92.40 versus 92.24 for equal averaging (Table B.5). These controls retain the feature branches while varying routing or fusion. With the auxiliary-list size fixed at 256 and the same loss weight, CoFiDrive trained with CSA reaches 92.40 PDMS, compared with 92.00 when the original proposals are repeated four times (Table B.5). Thus the reported gain is not reproduced by repeating the same candidates. The patch branch adds 2.01M parameters; CSA adds neither parameters nor inference candidates. These are within-table score differences, not estimates of statistical significance.

TC count sensitivity. Larger memories and more experts do not consistently improve PDMS (Table 4). Sixteen tokens per camera score 91.85, versus 91.66 for both 32 and 64. Six experts score 91.85, exceeding four and eight by 0.31 and 0.26 points. The default counts give the highest point estimate within each sweep. The 91.85 sweep default and the 92.40 sequential-ablation result belong to separate run groups. Their available summaries do not establish the source of the 0.55-point gap. We therefore interpret the count comparisons within each sweep, not across the two groups, and do not attribute the gap to seed variation.

4.5 Qualitative Analysis

At the intersection in Figure 2, CoFiDrive selects a trajectory closer to the ground-truth path. In the displayed attention maps (Figure 3), its registers attend to road geometry, vehicles, and traffic lights, while DrivoR also shows background activations. Appendix C presents additional maneuvers. These examples illustrate the behavior of the complete planners; they neither isolate the router nor quantify how much visual information is preserved.

5 Limitations

CoFiDrive uses the current four camera views and ego state rather than a history of visual observations. It therefore cannot directly observe the motion of surrounding agents across time or changes that occur while an agent is occluded. Inference is also limited to the 64 trajectories produced by the generator: the scorer can rank that set, yet it cannot recover a maneuver absent from the proposals.

The train85k main result is a mean over three independently seeded training runs; seed-level dispersion is not reported. The ablations report configuration-level scores without uncertainty estimates, so small differences should not be interpreted as statistically established effects. The PDMS comparisons and qualitative examples evaluate complete planners, not ranking error on a fixed, shared proposal set.

CSA retains the main score loss on the 64 generated proposals and adds a 256-entry auxiliary list containing those proposals and 192 slots. Its effective diversity depends on seed quality, perturbation bounds, and the bicycle-model approximation; invalid slots fall back to their seeds and can therefore duplicate them. The PDM oracle supplies component targets for these slots during training. At inference, CCS remains active, while CSA and PDM target computation are not used. CCS adds patch attention, and all TC experts remain active. We do not report latency, throughput, or peak memory, so the compact generator memory is not evidence of a measured hardware speedup. NAVSIM and NavHard provide benchmark evaluation rather than evidence of closed-loop, on-vehicle safety.

6 Conclusion

CoFiDrive refines a 64-token register memory for trajectory generation and gives CCS access to pooled patch features when scoring candidates. PDM-guided control-space augmentation (CSA) supplies additional scorer supervision while inference remains limited to the original 64 proposals. In the 10-epoch sequential ablation, PDMS rises from 91.70 to 92.40 as TC, Dual-Channel CCS, and CSA are added. The full model attains 92.4 and 92.9 PDMS on NAVSIM-v1 with train85k and trainval103k, respectively, and 48.0 EPDMS on NavHard. The NavHard result is 0.7 points above the DrivoR model reproduced under the same 10-epoch training and evaluation protocol.

AI USE STATEMENT

Generative AI tools assisted with English-language editing and translation, manuscript organization, feedback on experimental design and result interpretation, schematic figure preparation, LaTeX formatting, and source-code consistency checks. They were not used to generate experimental measurements. The authors reviewed all AI-assisted text, analysis, figures, and code-related suggestions against the implementation, experiment records, and cited sources, and take responsibility for the final content.

ETHICS STATEMENT

This work studies autonomous-driving planning in benchmark simulation. The reported results should not be interpreted as evidence of on-road safety or deployment readiness. Any deployment would require independent closed-loop, real-world, safety, and robustness validation.

REPRODUCIBILITY STATEMENT

Section 3 defines the architecture, routing, scorer training, gradient paths, and objective; Section 4 reports the data splits, 10-epoch training budget, evaluation protocol, comparisons, and ablations. Appendices A and B provide the metric definitions, training configuration, loss weights, and architectural hyperparameters.

REFERENCES

Wei Cao, Marcel Hallgarten, Tianyu Li, Daniel Dauner, Xunjiang Gu, Caojun Wang, Yakov Miron, Marco Aiello, Hongyang Li, Igor Gilitschenski, Boris Ivanovic, Marco Pavone, Andreas Geiger, and Kashyap Chitta. Pseudo-simulation for autonomous driving. In *Proceedings of the 9th Conference on Robot Learning*, volume 305 of *Proceedings of Machine Learning Research*, pp. 4709–4722. PMLR, 2025.

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, 2023.

Timothée Darcet, Maxime Oquab, Julien Mairal, and Piotr Bojanowski. Vision transformers need registers. In *International Conference on Learning Representations*, 2024.

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.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. An image is worth 16x16 words: Transformers for image recognition at scale. In *International Conference on Learning Representations*, 2021.

William Fedus, Barret Zoph, and Noam Shazeer. Switch transformers: Scaling to trillion parameter models with simple and efficient sparsity. *Journal of Machine Learning Research*, 23(120):1–39, 2022.

Lan Feng, Yang Gao, Éloi Zablocki, Quanyi Li, Wuyang Li, Sichao Liu, Matthieu Cord, and Alexandre Alahi. RAP: 3D rasterization augmented end-to-end planning. In *International Conference on Learning Representations*, 2026.

Ke Guo, Haochen Liu, Xiaojun Wu, Jia Pan, and Chen Lv. iPad: Iterative proposal-centric end-to-end autonomous driving. *IEEE Robotics and Automation Letters*, 11(10):11142–11149, 2026.

Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, Lewei Lu, Xiaosong Jia, Qiang Liu, Jifeng Dai, Yu Qiao, and Hongyang Li. Planning-oriented autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 17853–17862, 2023.

Andrew Jaegle, Felix Gimeno, Andy Brock, Oriol Vinyals, Andrew Zisserman, and Joao Carreira. Perceiver: General perception with iterative attention. In *Proceedings of the 38th International Conference on Machine Learning*, volume 139 of *Proceedings of Machine Learning Research*, pp. 4651–4664. PMLR, 2021.

Bo Jiang, Shaoyu Chen, Hao Gao, Bencheng Liao, Qian Zhang, Wenyu Liu, and Xinggang Wang. VADv2: End-to-end vectorized autonomous driving via probabilistic planning. In *International Conference on Learning Representations*, 2026.

Ellington Kirby, Alexandre Boulch, Yihong Xu, Yuan Yin, Gilles Puy, Éloi Zablocki, Andrei Bursuc, Spyros Gidaris, Renaud Marlet, Florent Bartoccioni, Anh-Quan Cao, Nermin Samet, Tuan-Hung Vu, and Matthieu Cord. Driving on registers. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 32058–32069, 2026.

Kailin Li, Zhenxin Li, Shiyi Lan, Yuan Xie, Zhizhong Zhang, Jiayi Liu, Zuxuan Wu, Zhiding Yu, and Jose M Alvarez. Hydra-MDP++: Advancing end-to-end driving via expert-guided hydra-distillation. *arXiv preprint arXiv:2503.12820*, 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*, 2026a.

Yongkang Li, Kaixin Xiong, Xiangyu Guo, Fang Li, Sixu Yan, Gangwei Xu, Lijun Zhou, Long Chen, Haiyang Sun, Bing Wang, Kun Ma, Guang Chen, Hangjun Ye, Wenyu Liu, and Xinggang Wang. ReCogDrive: A reinforced cognitive framework for end-to-end autonomous driving. In *International Conference on Learning Representations*, 2026b.

Zhenxin Li, Kailin Li, Shihao Wang, Shiyi Lan, Zhiding Yu, Yishen Ji, Zhiqi Li, Ziyue Zhu, Jan Kautz, Zuxuan Wu, Yu-Gang Jiang, and Jose M. Alvarez. Hydra-MDP: End-to-end multimodal planning with multi-target hydra-distillation. *arXiv preprint arXiv:2406.06978*, 2024.

Zhenxin Li, Nadine Chang, Wenhao Yao, Xinglong Sun, Zi Wang, Maying Shen, Jingde Chen, Jingyu Song, Kailin Li, Zuxuan Wu, Shiyi Lan, and Jose M. Alvarez. ZTRS: Zero-human demonstration end-to-end autonomous driving with trajectory scorer. In *European Conference on Computer Vision*, 2026c.

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

Joan Puigcerver, Carlos Riquelme, Basil Mustafa, and Neil Houlsby. From sparse to soft mixtures of experts. In *International Conference on Learning Representations*, 2024.

Yongming Rao, Wenliang Zhao, Benlin Liu, Jiwen Lu, Jie Zhou, and Cho-Jui Hsieh. DynamicViT: Efficient vision transformers with dynamic token sparsification. In *Advances in Neural Information Processing Systems*, volume 34, pp. 13937–13949, 2021.

Carlos Riquelme, Joan Puigcerver, Basil Mustafa, Maxim Neumann, Rodolphe Jenatton, André Susano Pinto, Daniel Keysers, and Neil Houlsby. Scaling vision with sparse mixture of experts. In *Advances in Neural Information Processing Systems*, volume 34, pp. 8583–8595, 2021.

Michael S. Ryoo, AJ Piergiovanni, Anurag Arnab, Mostafa Dehghani, and Anelia Angelova. TokenLearner: Adaptive space-time tokenization for videos. In *Advances in Neural Information Processing Systems*, volume 34, pp. 12786–12797, 2021.

Noam Shazeer, Azalia Mirhoseini, Krzysztof Maziarz, Andy Davis, Quoc Le, Geoffrey Hinton, and Jeff Dean. Outrageously large neural networks: The sparsely-gated mixture-of-experts layer. In *International Conference on Learning Representations*, 2017.

Chen Shi, Shaoshuai Shi, Kehua Sheng, Bo Zhang, and Li Jiang. DriveX: Omni scene modeling for learning generalizable world knowledge in autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 28599–28609, 2025.

Ziying Song, Lin Liu, Hongyu Pan, Bencheng Liao, Mingzhe Guo, Lei Yang, Yongchang Zhang, Shaoqing Xu, Caiyan Jia, and Yadan Luo. DIVER: Reinforced diffusion breaks imitation bottlenecks in end-to-end autonomous driving. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, pp. 1–17, 2026. Early Access.

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

Yuqi Wang, Xinghang Li, Wenxuan Wang, Junbo Zhang, Yingyan Li, Yuntao Chen, Xinlong Wang, and Zhaoxiang Zhang. Unified vision-language-action model. In *International Conference on Learning Representations*, 2026.

Xinshuo Weng, Boris Ivanovic, Yan Wang, Yue Wang, and Marco Pavone. PARA-Drive: Parallelized architecture for real-time autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 15449–15458, 2024.

Maciej Wozniak, Lianhang Liu, Yixi Cai, and Patric Jensfelt. PRIX: Learning to plan from raw pixels for end-to-end autonomous driving. *IEEE Robotics and Automation Letters*, 11(5):6400–6407, 2026.

Penghao Wu, Xiaosong Jia, Li Chen, Junchi Yan, Hongyang Li, and Yu Qiao. Trajectory-guided control prediction for end-to-end autonomous driving: A simple yet strong baseline. In *Advances in Neural Information Processing Systems*, volume 35, pp. 6119–6132, 2022.

Yihong Xu, Eloi Zablocki, Yuan Yin, Elias Ramzi, Ellington Kirby, Alexandre Boulch, and Matthieu Cord. Test-time trajectory optimization for autonomous driving. *arXiv preprint arXiv:2606.07170*, 2026.

Wenhao Yao, Zhenxin Li, Shiyi Lan, Zi Wang, Xinglong Sun, Jose M. Alvarez, and Zuxuan Wu. DriveSuprim: Towards precise trajectory selection for end-to-end planning. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 11910–11918, 2026. Issue 14.

Chengran Yuan, Zhanqi Zhang, Jiawei Sun, Shuo Sun, Zefan Huang, Christina Dao Wen Lee, Dongen Li, Yuhang Han, Anthony Wong, Keng Peng Tee, and Marcelo H. Ang, Jr. DRAMA: An efficient end-to-end motion planner for autonomous driving with Mamba. In *International Symposium on Robotics Research*, 2024.

Yupeng Zheng, Pengxuan Yang, Zebin Xing, Qichao Zhang, Yuhang Zheng, Yinfeng Gao, Pengfei Li, Teng Zhang, Zhongpu Xia, Peng Jia, XianPeng Lang, and Dongbin Zhao. World4Drive: End-to-end autonomous driving via intention-aware physical latent world model. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 28632–28642, 2025.

Yanqi Zhou, Tao Lei, Hanxiao Liu, Nan Du, Yanping Huang, Vincent Zhao, Andrew M. Dai, Zhifeng Chen, Quoc V. Le, and James Laudon. Mixture-of-experts with expert choice routing. In *Advances in Neural Information Processing Systems*, volume 35, pp. 7103–7114, 2022.

Zewei Zhou, Tianhui Cai, Seth Zhao, Yun Zhang, Zhiyu Huang, Bolei Zhou, and Jiaqi Ma. AutoVLA: A vision-language-action model for end-to-end autonomous driving with adaptive reasoning and reinforcement fine-tuning. In *Advances in Neural Information Processing Systems*, volume 38, pp. 31725–31761, 2025.

CoFiDrive: Scene-Conditioned Registers and Coarse-to-Fine Trajectory Scoring for End-to-End Driving

Supplementary Materials

A Benchmark and Metrics

Bibliographic details for citations in this supplement are listed in the main paper.

Data and evaluation. NAVSIM (Dauner et al., 2024) uses curated OpenScene scenes derived from nuPlan and sampled at 2 Hz. Each frame provides surround-view cameras, LiDAR sweeps, HD maps, and dynamic-object annotations. CoFiDrive uses the four camera views and current ego state; the other benchmark data are used for evaluation and oracle labeling, not as inputs to the neural planner at inference. NAVSIM-v1 evaluates predicted trajectories by a non-reactive rollout, not by repeated observation–action interaction with a live environment.

NAVSIM-v1. Each 4 s plan is executed at 10 Hz by a kinematic bicycle model with an LQR controller, while surrounding agents replay their recorded trajectories. PDMS multiplies the safety terms in $\mathcal{M}$ by a weighted average of the quality terms in $\mathcal{A}$. All terms lie in $[0, 1]$; a multiplicative term therefore scales the entire score:

$$\mathrm{PDMS} = \prod_{m \in \mathcal{M}} s_m \cdot \frac{\sum_{a \in \mathcal{A}} w_a \, s_a}{\sum_{a \in \mathcal{A}} w_a}, \tag{A.1}$$

with multiplicative set $\mathcal{M} = \{\mathrm{NC}, \mathrm{DAC}\}$ and additive set $\mathcal{A} = \{\mathrm{TTC}, \mathrm{EP}, \mathrm{C}\}$.

NAVSIM-v2. NavHard uses NAVSIM-v2's two-stage pseudo-simulation protocol (Cao et al., 2025). Stage 1 evaluates original observations with reactive traffic; Stage 2 evaluates synthetic continuations. Human-reference filtering applies to an expanded metric set with rule compliance, lane keeping, and history and extended comfort. For each evaluated observation, the component scores form

$$q_{\mathrm{obs}} = \prod_{m \in \mathcal{M}'} s_m \cdot \frac{\sum_{a \in \mathcal{A}'} w_a \, s_a}{\sum_{a \in \mathcal{A}'} w_a}, \tag{A.2}$$

where $\mathcal{M}' = \{\mathrm{NC}, \mathrm{DAC}, \mathrm{DDC}, \mathrm{TLC}\}$ and $\mathcal{A}' = \{\mathrm{TTC}, \mathrm{EP}, \mathrm{LK}, \mathrm{HC}, \mathrm{EC}\}$. Each s is a benchmark-filtered subscore. The final NavHard EPDMS multiplies the Stage 1 score by the proximity-weighted Stage 2 score. Table A.1 lists metric roles and weights. Reported PDMS and EPDMS use the 0–100 scale. Scores average scene-level evaluations; applying Eq. (A.1) to component means does not, in general, recover mean PDMS.

B Supplementary Implementation Details

Training budget. CoFiDrive, the DrivoR baseline, and all ablations use 10 epochs. This budget covers both NAVSIM-v1 splits, `navtrain` 85k and trainval 103k, and NavHard, where CoFiDrive and DrivoR share training and evaluation splits. Only the train85k main comparison averages three independently seeded training runs; this aggregation does not apply to trainval103k, NavHard, or the ablation tables. Cited results retain their published protocols. Table B.1 lists optimization and control-space augmentation (CSA) settings. CCS denotes the Cross-Compression Scorer, which reads both scene and patch memories; CSA changes only its training candidate list.

Table A.1: **Components of PDMS and EPDMS.** Metrics marked with • enter multiplicatively as gating penalties; the remaining metrics enter additively with the listed weight, normalized by $\sum_a w_a$. A dash denotes a metric that is not used by that score. The reported benchmark score is the average of scene-level scores, not the score computed from average components.

Metric	Role	PDMS	EPDMS
NC	no at-fault collision	•	•
DAC	drivable-area compliance	•	•
DDC	driving-direction compliance	–	•
TLC	traffic-light compliance	–	•
TTC	time-to-collision margin	5	5
EP	ego progress	5	5
C	comfort	2	–
LK	lane keeping	–	2
HC	history comfort	–	2
EC	extended comfort	–	2

Table B.1: **Training settings.** All models trained in this work use 10 epochs. Noise scales refer to acceleration and yaw rate.

Setting	Value
Training split	`navtrain` 85k / trainval 103k
Optimizer	AdamW
Base learning rate	2×10^{-4}
Batch size per GPU	16
GPUs	4
Epochs (all runs)	10
Augmentation seeds (top-K)	16
Augmentation slots/seed	12
Auxiliary candidates	256
Accel. / yaw-rate noise	0.3 / 0.1

Table B.2: **Default architecture.** Counts are per model unless stated otherwise; each proposal contains eight future poses.

Setting	Value
Backbone	DINOv2 ViT-S/14 (reg4)
LoRA rank	32
Cameras	4
Registers/camera	16
Scene tokens	64
Hidden width	256
Experts / active	6 / 6 (all)
Queries / proposals	64 / 64
Decoder / scorer layers	4 / 4
Horizon / frequency	4 s / 2 Hz
Patch-pool grid/view	12×21

Loss function. The objective combines trajectory regression, main and auxiliary score supervision, and two router regularizers for the semantic token compressor (TC):

$$\begin{aligned}\mathcal{L} = \lambda_{\text{traj}}\mathcal{L}_{\text{traj}} + \lambda_{\text{main}}\mathcal{L}^{\text{main}}_{\text{score}} + \lambda_{\text{aux}}\mathcal{L}^{\text{aux}}_{\text{score}} \\ + \lambda_{\text{lb}}\mathcal{L}_{\text{lb}} + \lambda_{\text{ent}}\mathcal{L}_{\text{ent}}.\end{aligned} \tag{B.1}$$

Here $\mathcal{L}_{\text{traj}}$ is best-of-64 L1 regression on the final proposals. Proposals are also produced before and after each of four decoder blocks. Intermediate regression losses have zero weight. For the 103k setting, a longer reference trajectory is cubic-spline resampled to eight poses and supplies a second best-of-64 term:

$$\mathcal{L}^{\text{103k}}_{\text{traj}} = \min_j d(\tau_j, \tau_{\text{gt}}) + \min_j d(\tau_j, \tau_{\text{long}}). \tag{B.2}$$

The two minima are independent and may select different proposals. With k additional future poses, the source horizon is $4 + 0.5k$ seconds. The implementation interpolates each of (x, y, θ) at indices $q_i = i + k(i+1)(i+2)/(T(T+1))$, $i = 0, \ldots, T-1$, with $T = 8$ and zero-based source indices. If the longer reference is unavailable, it reuses the ordinary target. This target changes supervision, not the eight-pose inference output. Both score terms sum six binary cross-entropy losses with unit head weights: $\mathcal{L}^{\text{main}}_{\text{score}}$ supervises the original 64 proposals; $\mathcal{L}^{\text{aux}}_{\text{score}}$ supervises all 256 auxiliary entries, including the originals. $\mathcal{L}_{\text{lb}}$ penalizes uneven batch-average router probabilities. Minimizing the negative mean routing entropy $\mathcal{L}_{\text{ent}}$ encourages higher entropy. All six experts remain active. For batch size B and routing probabilities $p_{b,e}$, the regularizers are

$$\begin{aligned}\mathcal{L}_{\text{lb}} &= E \sum_{e=1}^{E} \left(\frac{1}{B} \sum_{b=1}^{B} p_{b,e} \right)^2, \\ \mathcal{L}_{\text{ent}} &= \frac{1}{B} \sum_{b,e} p_{b,e} \log(p_{b,e} + 10^{-8}).\end{aligned} \tag{B.3}$$

Tables B.2 and B.3 list architecture settings and objective weights, respectively.

Table B.3: **Training objective weights.** Coefficients in Eq. (B.1). Both score losses sum six component losses; auxiliary supervision includes the original and augmented candidates.

Term	Weight	Value
Trajectory	λ_{traj}	1.0
Main score	λ_{main}	1.0
Auxiliary score	λ_{aux}	0.5
Load balance	λ_{lb}	0.01
Router entropy	λ_{ent}	0.001

Component targets and inference scores. The six heads predict no at-fault collision (NC, labeled NOC in the code), DAC, TTC, EP, DDC, and comfort. For NC and DDC, target values of 0.5 are mapped to zero before binary cross-entropy. The TTC loss excludes entries with the invalid-target sentinel 2.0 and normalizes by the number of valid entries, clamped to at least one. Other heads use mean binary cross-entropy. These label transformations belong to scorer training; benchmark metrics are evaluated according to Appendix A.

Let $\hat{s}_m$ denote the sigmoid output of component head m. The configured ranking score is evaluated in log space; its dependence on the supplied proposal set, visual memories, and ego state is suppressed below:

$$\begin{aligned} r_\tau &= 5\big(\hat{s}_{\text{TTC}} + \hat{s}_{\text{EP}}\big) + 2\hat{s}_{\text{C}}, \\ S(\tau) &= \log \hat{s}_{\text{NC}} + \log \hat{s}_{\text{DAC}} \\ &\quad + \log\big(\max\{r_\tau, 10^{-6}\}\big)\,. \end{aligned} \tag{B.4}$$

The implementation uses log-sigmoid for the multiplicative terms. DDC is supervised during training and omitted from the inference aggregation. The additive normalizer is constant across candidates and does not change the argmax. This score ranks proposals; it is neither an oracle PDMS measurement nor a prediction of all NAVSIM-v2 components.

CSA control perturbations. The PDM oracle scores the 64 proposals and selects the top 16 as augmentation seeds. For each seed, an approximate kinematic projection estimates acceleration and yaw rate from successive poses at $\Delta t = 0.5\,\text{s}$. We allocate 12 slots per seed and independently perturb each control channel $\kappa \in \{a, \omega\}$. Suppressing seed and slot indices, the noise is

$$\begin{aligned} h_1 &= \epsilon_1, \\ h_t &= \rho h_{t-1} + \sqrt{1-\rho^2}\epsilon_t \quad (t \geq 2), \\ b &\sim \mathcal{N}(0, (0.7\sigma_\kappa)^2), \\ \delta c_t &= b + \frac{0.3\sigma_\kappa h_t}{\max(\text{std}_{1:T}(h), 10^{-4})}, \end{aligned} \tag{B.5}$$

where the innovations ϵ_t are independent standard Gaussian draws, $\text{std}_{1:T}$ is the sample standard deviation, $\rho = 0.8$, $\sigma_a = 0.3\,\text{m}\,\text{s}^{-2}$, and $\sigma_\omega = 0.1\,\text{rad}\,\text{s}^{-1}$. The perturbed acceleration is clipped to $[-4.05, 2.40]\,\text{m}\,\text{s}^{-2}$ and yaw rate to $[-0.95, 0.95]\,\text{rad}\,\text{s}^{-1}$. The bicycle model then rolls out the clipped sequence from the current longitudinal speed.

A rollout must maintain velocity at least $-0.2\,\text{m}\,\text{s}^{-1}$. At every pose, its absolute y-coordinate deviation in the current ego frame must not exceed $2.0\,\text{m}$; Euclidean position and absolute heading deviations from its seed must not exceed $6.0\,\text{m}$ and $0.8\,\text{rad}$, respectively. Each slot retains its first accepted sample in at most four attempts. Unresolved slots retain the seed without reapplying these checks. The resulting 192 entries are concatenated with the 64 originals. They may include duplicates and are not guaranteed to satisfy collision or drivable-area constraints; the PDM oracle supplies the component labels for the retained trajectories.

Architecture. The four camera streams share DINOv2 ViT-S/14 with Q/V LoRA adapters of rank 32. Each 1148×672 image produces a 48×82 patch grid of width 384. Sixteen added registers per camera form the compact path. All six SwiGLU experts have hidden width 1024, dropout 0.1,

and scene-level soft routing with training logit jitter of standard deviation 0.1. The residual scale is initialized to one. For CCS, detached patches are pooled per view and projected by a $384 \to 256 \to 256$ MLP. Two-dimensional position, camera, and token-type embeddings are added before layer normalization. Candidate coordinates use a $24 \to 1024 \to 256$ embedding MLP.

B.1 Additional Ablations

All controls use train85k, navtest, 10 epochs, 64 scene tokens, and 64 inference candidates. In Table B.5, "original $\times 4$" repeats the 64 proposals to form a 256-entry auxiliary list.

Table B.4: **Register processing.** Dual-Channel CCS and CSA fixed; train85k/navtest PDMS.

Register processing	PDMS ↑
Bypass TC	92.05
Single SwiGLU	92.12
Uniform 6-expert mix	92.18
Scene-routed 6-expert mix	**92.40**

Table B.5: **Fusion and supervision.** TC and both memories fixed; train85k/navtest PDMS.

Fusion / auxiliary candidates	PDMS ↑
Gate / none	91.90
Average / control-space aug. + original	92.24
Gate / original $\times 4$	92.00
Gate / control-space aug. + original	**92.40**

Control interpretation. Table B.4 compares register transformations with both CCS memories and CSA fixed; Table B.5 varies the fusion rule or auxiliary candidates while retaining TC and both memories. Their conclusions are summarized in the main-paper experiments section. They do not isolate ranking accuracy on identical generator outputs or establish the statistical significance of the score gaps.

C Qualitative Visualization

Figures C.1–C.3 compare matched straight, left-turn, and right-turn scenes from NAVSIM-v1 `navtest`. Each planner generates and scores its own proposals. In these cases, CoFiDrive follows the lane center or visible turn geometry more closely; DrivoR sometimes cuts inside a turn or approaches an adjacent lane. Figure C.4 shows register attention, and Figure C.5 shows further proposal selections.

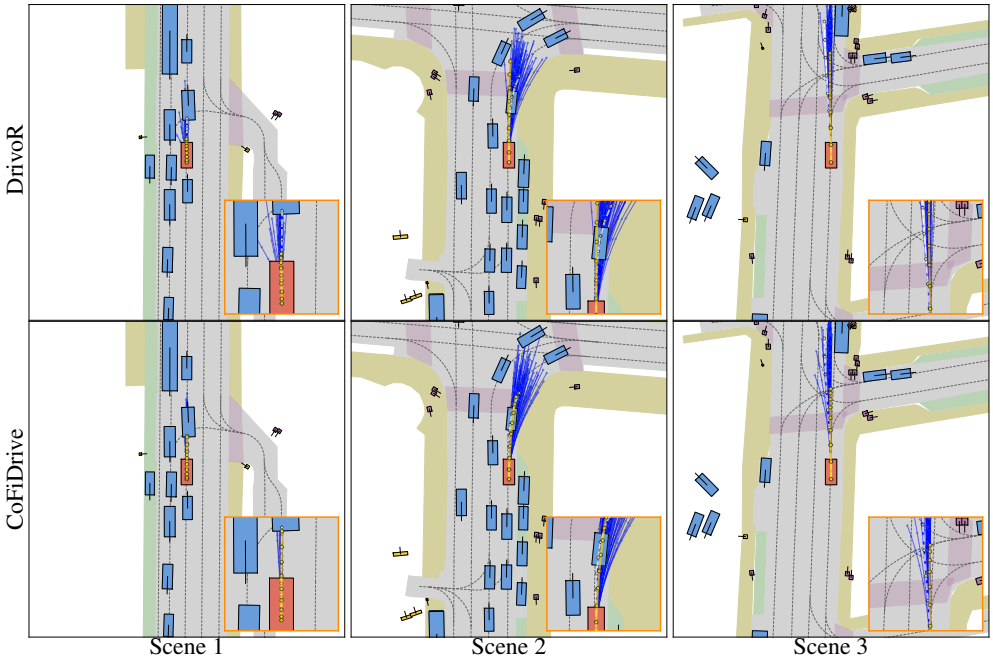

Figure C.1: **Straight driving on NAVSIM-v1 navtest.** Matched scenes; DrivoR is above CoFiDrive. Blue denotes proposals and yellow the selected path. CoFiDrive stays closer to the lane center in these cases.

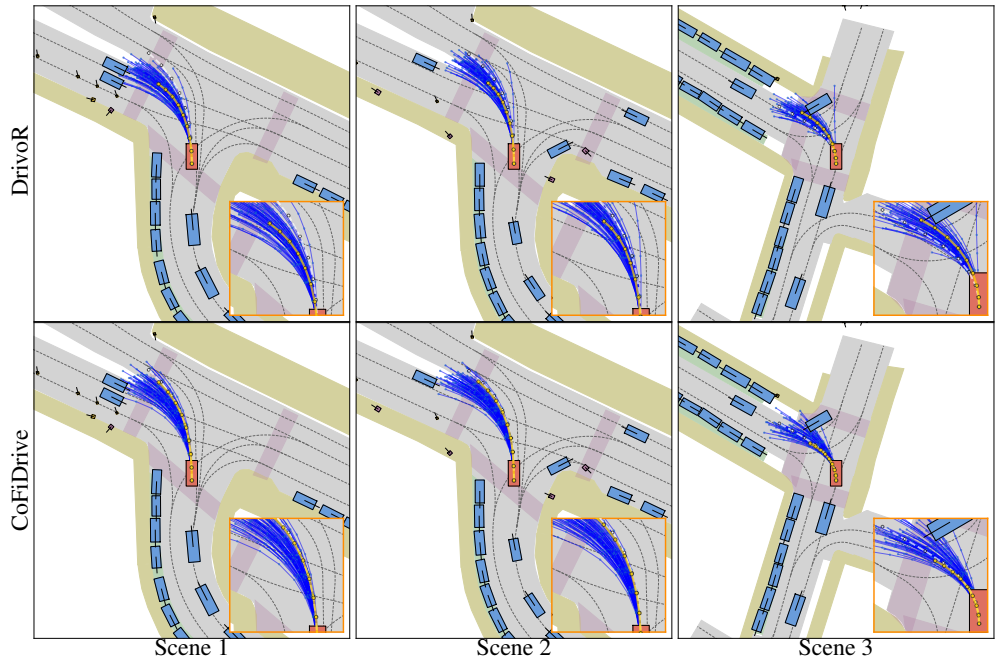

Figure C.2: **Left turns on NAVSIM-v1 navtest.** Matched scenes compare DrivoR (top) and CoFiDrive (bottom). Blue denotes proposals and yellow the selected path. CoFiDrive follows the visible turning-lane geometry more closely in these cases.

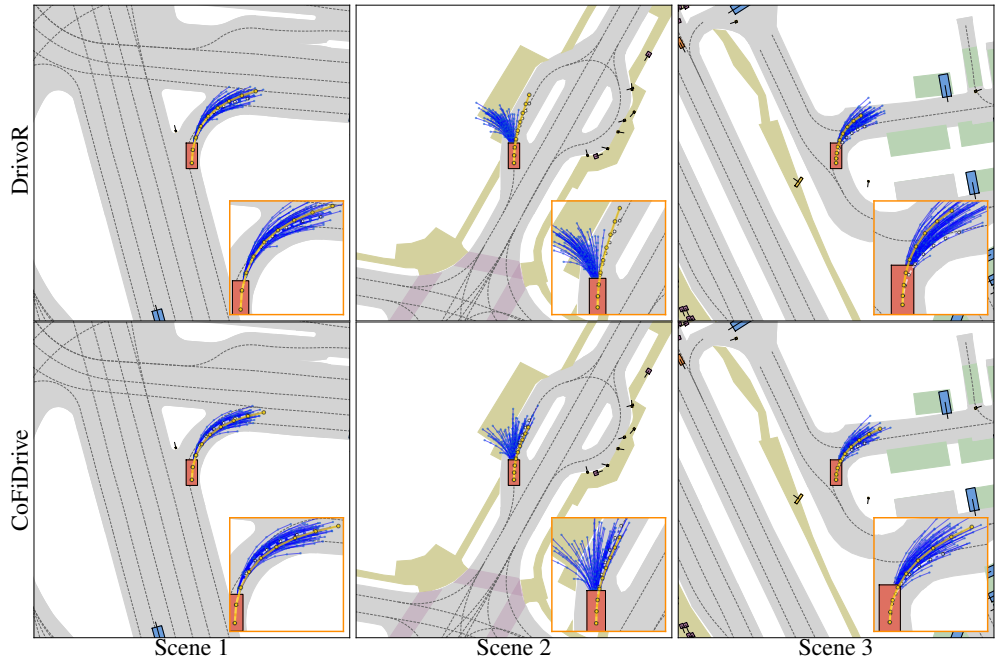

Figure C.3: **Right turns on NAVSIM-v1 navtest.** Matched scenes range from a single-lane turn to a channelized intersection. DrivoR is above CoFiDrive; blue curves denote proposals and yellow waypoints the selection. In these cases, CoFiDrive follows the visible turn lane more closely.

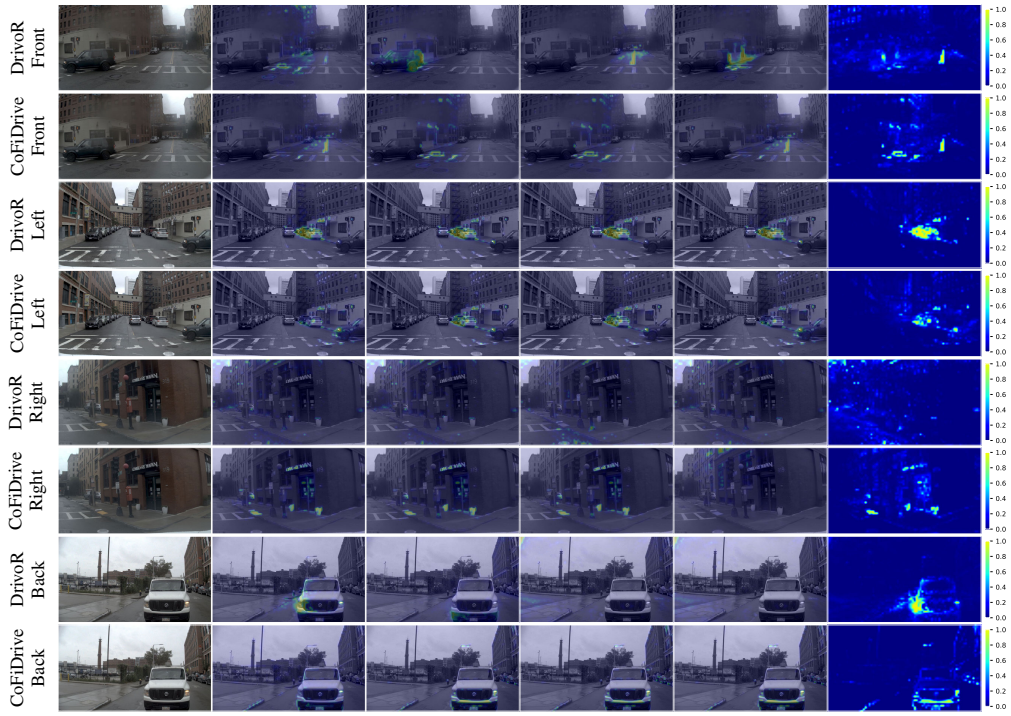

Figure C.4: **Surround-view register attention.** For each camera, paired rows show DrivoR (above) and CoFiDrive (below). Columns show RGB, attention maps for registers 0, 4, 8, and 12, and an accompanying mean map.

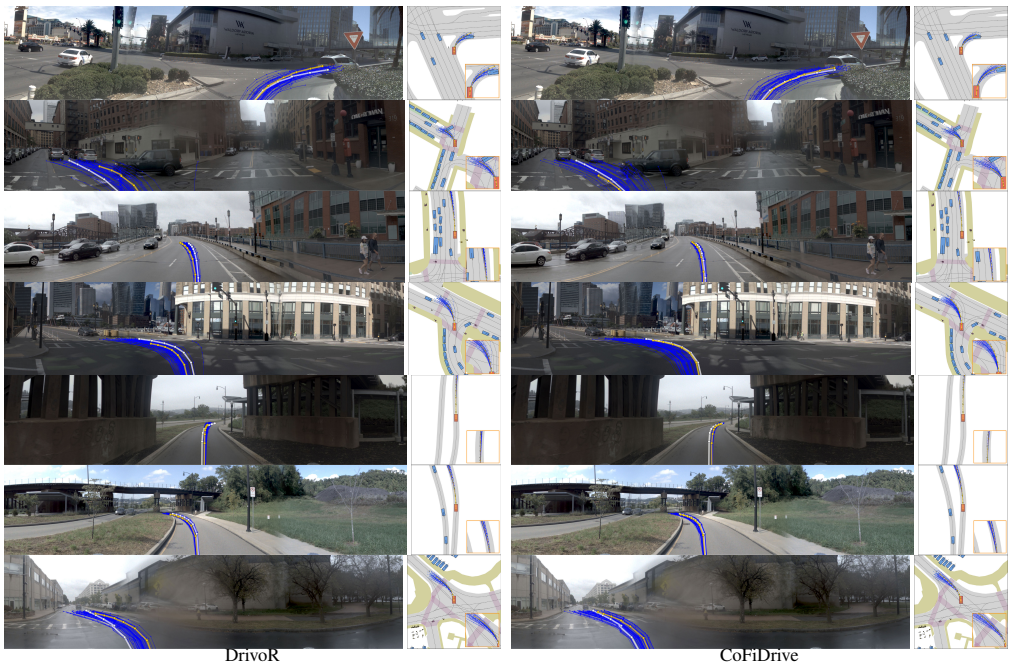

Figure C.5: **Candidate generation and selection across maneuvers.** Rows pair DrivoR (left) and CoFiDrive (right) on NAVSIM-v1 navtest merges and intersection turns. Blue marks proposals, yellow the selected path, and white the ground truth. Each planner scores its own proposals; the panels show complete planner outputs.