FOCUSDRIVE: REASONING WITH VISUAL FOCUS FOR AUTONOMOUS DRIVING

Anonymous authors
Paper under double-blind review

ABSTRACT

Driving decisions depend on both where to focus and how to act on what is seen. Effective driving reasoning must establish which objects matter, where they are, and how they inform the intended action. Text-based rationales can describe a driving response while leaving its correspondence to specific visual evidence implicit. Visual focus provides a concrete starting point for this connection by identifying what matters in the scene and where it is. We propose **FocusDrive**, a structured multimodal reasoning framework that organizes end-to-end planning around explicit visual focus. It pairs descriptions of decision-relevant objects with image-patch references, bringing explicit visual focus into the reasoning that generates driving plans and trajectories. We first assess this focus representation through driver gaze prediction, then investigate its role in planning reasoning using driving-focus annotations within existing NAVSIM training scenes. Experiments on W3DA and NAVSIM demonstrate competitive gaze prediction and end-to-end planning performance, with FocusDrive improving over text-based chain-of-thought. These results support visual focus as an effective link between scene understanding and driving action.

1 INTRODUCTION

Knowing where to focus is an essential part of driving. Visual focus concerns the scene elements selected for consideration as a driver decides how to respond. Human gaze offers an observable perspective on this process, motivating a substantial body of research on driver attention prediction. Early deep saliency models learned spatial attention distributions from convolutional features (Cornia et al., 2016), while driving-specific approaches incorporated temporal context, critical-event sampling, and semantic scene structure to predict attention in routine and challenging situations (Palazzi et al., 2019; Xia et al., 2018; Fang et al., 2022). Predicted attention has also been used to improve learned driving control (Makrigiorgos et al., 2019). Together, these studies establish a foundation for modeling driving focus and understanding how drivers prioritize visual information.

More recent work connects these spatial predictions to language, making the objects and reasons behind attention explicit. LLada jointly models where drivers attend, what they attend to, and why (Zhou et al., 2025a); FSDAM couples attention prediction with structured explanations under limited supervision (Hamid et al., 2026); and DualGaze-VLM studies text-guided object-level gaze prediction (Ke et al., 2026). Together, gaze prediction and its language-based explanations connect where drivers look with the objects and reasons behind that selection. This perspective on human attention motivates our approach to driving reasoning: explicitly identify and locate the relevant scene elements, then reason about how to respond to them.

Intermediate reasoning has become an important means of connecting scene understanding with action generation in driving VLMs and VLAs. Language-based approaches organize scene interpretation into descriptions and structured questions (Sima et al., 2025; Tian et al., 2025; Hwang et al., 2025), and further developments connect CoT reasoning with action generation (Zhou et al., 2025b; Chen et al., 2026). In current driving VLA CoT, however, intermediate reasoning is primarily expressed in natural language. Object descriptions convey what matters and why, but their correspondence to specific regions of the input image remains implicit in the generated sequence. Meanwhile, recent VLM research explores richer spatial and multimodal representations to improve visual understanding, ranging from textual coordinate prediction (Song et al., 2026) to continuous

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.

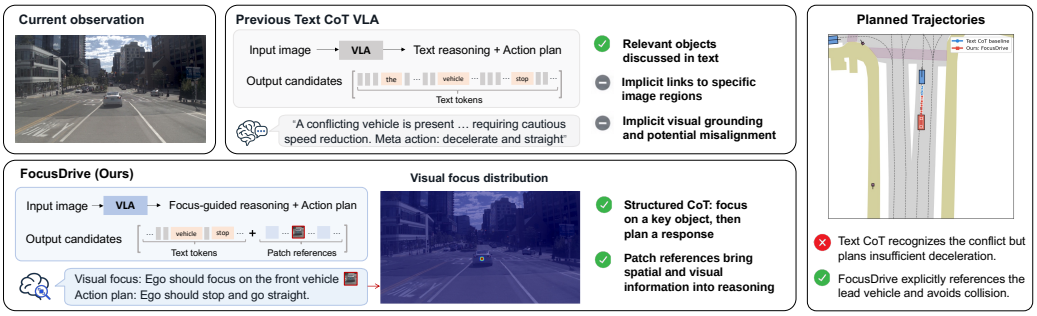

Figure 1: Compared with text-only CoT, FocusDrive explicitly links objects to image patches and structures reasoning around identifying and locating a key object, then determining the appropriate driving response.

visual tokens (Qin et al., 2026) and image-region references (Su et al., 2026). These developments offer ways to connect language reasoning more directly with the spatial structure and visual information of an image. We bring this perspective into driving through visual focus: connecting the object under consideration to its image representation before reasoning about the driving response.

We introduce **FocusDrive**, a structured multimodal CoT framework that organizes driving reasoning around explicit visual focus. FocusDrive pairs an object description with a generated patch reference to express decision-relevant visual focus within driving reasoning, using PaDT's dynamic visual-reference interface (Su et al., 2026). The description identifies the scene element under consideration, while the reference connects it to an image region and its visual representation. This pairing expresses where the model focuses and provides image-grounded context for the reasoning that follows. We first apply this formulation to language-related driver gaze prediction, where a lightweight adapter converts patch scores into dense attention heatmaps. This task evaluates how well the representation captures the spatial distribution of driver focus. For end-to-end planning, the policy generates a visual-focus span, an action plan, and future waypoints in sequence. Its object description and patch reference supply spatial and visual context as it reasons about how to respond to the selected scene element, connecting visual focus with action generation.

To support learning this form of reasoning, we develop driving-focus annotations within NAVSIM. Automated risk assessment (Wang et al., 2015; Cheng et al., 2026) and VLM-based screening identify candidate scenes and objects, concentrating focus annotation on situations that call for it, inspired by AdaThinkDrive's selective reasoning perspective (Luo et al., 2025). The annotators refine these candidates and select the final visual focus, bringing human judgments of driving relevance into the spatial supervision. These annotations make human-selected driving focus learnable within planning sequences, pairing object descriptions with spatial references before the subsequent reasoning and action targets.

We evaluate FocusDrive on W3DA (Zhou et al., 2025a) and NAVSIM (Dauner et al., 2024), examining both its ability to predict driver gaze and its effectiveness in end-to-end planning. Competitive gaze prediction across normal, safety-critical, and accident scenarios demonstrates the framework's ability to model the spatial distribution of driver attention. We further evaluate the use of explicit visual focus within planning CoT, where FocusDrive achieves competitive end-to-end planning performance on NAVSIM. Its improvement over text-based CoT supports the value of explicitly incorporating driving focus into multimodal reasoning. Ablations further examine how focus representation and visual-reference learning contribute to this improvement.

Our contributions are threefold:

- We propose FocusDrive, a structured multimodal CoT framework that organizes driving reasoning around explicit visual focus, connecting object identification and localization with response reasoning and end-to-end planning.
- We develop driving-focus annotations within NAVSIM, capturing human selections of decision-relevant objects and their image locations to support learning focus within planning reasoning.

- We demonstrate competitive performance in driver gaze prediction and end-to-end planning, and show that visual-focus reasoning improves planning over text-based CoT with focus annotations on a small subset of existing NAVSIM training scenes.

2 Related Work

Driver attention and language-guided gaze prediction. Learning where drivers look has long provided a route to understanding visual selection in driving. Convolutional saliency models (Cornia et al., 2016) learn dense spatial predictions from visual features. Driving-specific studies extend this line to temporal scene context in DR(eye)VE (Palazzi et al., 2019), critical situations in BDD-A (Xia et al., 2018), and accident scenarios in DADA (Fang et al., 2022). The usefulness of predicted human attention for learned driving control has also been explored (Makrigiorgos et al., 2019). More recent approaches connect spatial attention to language: LLada jointly predicts attention maps, attended objects, and explanations (Zhou et al., 2025a); FSDAM learns spatial prediction and structured explanations with limited semantic supervision (Hamid et al., 2026); and DualGaze-VLM introduces text-guided object-level gaze modeling alongside global prediction (Ke et al., 2026). Together, these works establish a progression from predicting attention distributions to describing their semantic content.

Reasoning in driving vision-language-action models. Research on reasoning in driving VLAs addresses both how scene information is represented and how it is used to produce actions. DriveLM (Sima et al., 2025), DriveVLM (Tian et al., 2025), and EMMA (Hwang et al., 2025) organize scene understanding and planning through language. RDA-Driver (Huang et al., 2025) and ORION (Fu et al., 2025) study the connection between reasoning and action generation, while AutoVLA (Zhou et al., 2025b) and AdaThinkDrive (Luo et al., 2025) adapt the amount of reasoning to the driving situation. A further line of work enriches the information carried by intermediate representations. CoT-VLA (Zhao et al., 2025) uses future visual goals in robotic manipulation, and FSDrive (Zeng et al., 2025) introduces future visual scenes into driving CoT. DriveVLA-W0 (Li et al., 2026a) exploits world-model supervision, OneVL (Lu et al., 2026) develops latent reasoning for planning, and DynVLA (Shang et al., 2026) reasons through compact dynamics tokens. DriveTeach-VLA (Yang et al., 2026) uses spatial trajectory prompts to guide policy learning. Together, these approaches establish intermediate reasoning as a means of coupling scene interpretation with action generation, through both language and increasingly rich visual representations.

3 Method

FocusDrive organizes driving reasoning around explicitly selected visual focus. The model describes an object, references its location in the image, and continues reasoning toward a driving response with the selected visual features in context. This formulation connects what the model considers with how it plans to act. We describe the construction of driving-focus supervision and the visual-reference mechanism, then present its use in gaze prediction and end-to-end planning.

3.1 Task Formulation

Driver gaze prediction. Given a driving image I and a task prompt q, the goal is to predict a dense gaze map $\hat{G} \in [0,1]^{H_g \times W_g}$ that matches the recorded driver gaze G. We study global attention prediction on W3DA (Zhou et al., 2025a). The language and spatial annotations associated with the training examples provide supervision for relating attended objects to image regions, while the gaze maps describe how attention is distributed over the scene. The prompt contains a scene description and task instruction.

End-to-end planning. The NAVSIM planning task (Dauner et al., 2024) provides sensor observations together with a navigation command and ego state. We denote the planning input by $x = (\mathcal{O}, c, s)$, where $\mathcal{O}$ contains the available sensor observations, c is the navigation command, and s is the ego state. The output is an ego-frame future trajectory $\tau = \{(x_k, y_k, \theta_k)\}_{k=1}^{K}$, where θ_k denotes heading. Under the NAVSIM evaluation protocol considered here, the trajectory consists of $K = 8$ waypoints spanning four seconds at 0.5-second intervals.

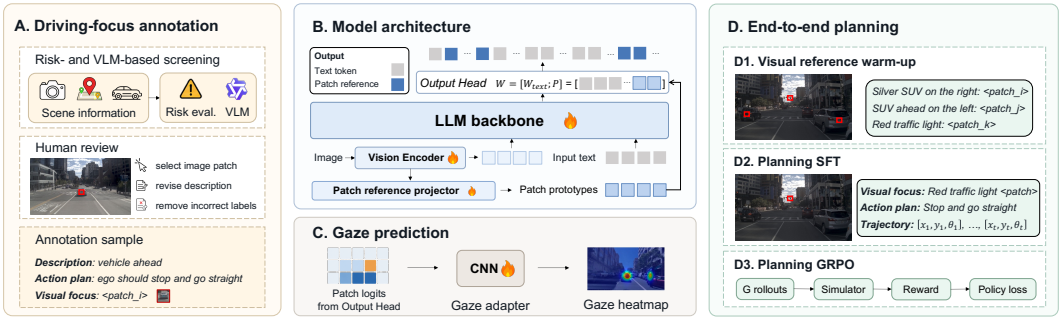

Figure 2: FocusDrive overview. Risk- and VLM-based screening followed by human review provides driving-focus annotations. Image-patch references support gaze prediction through an adapter and structured focus-to-action reasoning for planning, trained with reference warm-up, SFT, and PDMS-based GRPO.

3.2 Driving-Focus Annotations

Gaze data provides a foundation for learning where drivers attend. To make visual focus useful for planning, we additionally need spatial annotations that relate scene elements to the driving decision. We construct these driving-focus annotations within the existing NAVSIM training scenes. Each annotation pairs an object description d_o with a human-selected image location $u = (u_x, u_y) \in [0, 1]^2$, expressing both the content and location of the selected focus.

The annotation process proceeds from coarse screening to human selection. Drawing on established approaches to driving risk assessment (Hayward, 1972; Wang et al., 2015; Cheng et al., 2026), we screen for potentially critical agent interactions and combine these physical risk cues with VLM assessments of the scene and its objects. Together, they narrow the scenes and candidate objects requiring human consideration. Inspired by the selective reasoning perspective of AdaThinkDrive (Luo et al., 2025), we concentrate explicit focus annotation on situations in which selecting a particular object helps explain the driving decision. The annotators then assess the candidates in context, choose the object to focus on, and manually mark its image location. The automated stage directs annotation effort, while human judgment determines the final spatial target.

We map each selected location to the VLM's spatial grid and pair its patch reference with the object description to form a focus span. Annotated scenes include this span before the driving plan, while all scenes retain plan and waypoint supervision. Annotation details appear in Appendix A.3.1.

3.3 Representing Visual Focus

FocusDrive expresses visual focus through an object description paired with a generated patch reference. Each reference points to a merged cell in the image grid and carries a visual representation contextualized by the vision encoder.

We implement image-patch references in the VLM using the dynamic visual-reference mechanism of PaDT (Su et al., 2026). For an input image I, the visual backbone produces spatial embeddings $E(I) = [e_1, \ldots, e_N]^\top \in \mathbb{R}^{N \times d}$, where N is the number of spatial reference units and d is the language model's hidden dimension. We refer to these grid cells as image patches.

A visual-reference projector transforms each patch embedding e_j into a prototype p_j in the VLM's hidden space. It first applies layer normalization, then adds a low-rank residual projection implemented by two bias-free linear layers with a low-dimensional bottleneck. The resulting prototypes form $P(I) = [p_1, \ldots, p_N]^\top \in \mathbb{R}^{N \times d}$. Each prototype serves both as an output choice and as the input embedding of its reference token.

Let $\mathcal{V}_{\text{text}}$ be the ordinary text vocabulary, with output embeddings $W_{\text{text}} \in \mathbb{R}^{|\mathcal{V}_{\text{text}}| \times d}$. We extend its output choices with the image-dependent reference set $\mathcal{V}_{\text{vis}}(I) = \{v_1, \ldots, v_N\}$. At decoding

step t, the hidden state $h_t \in \mathbb{R}^d$ scores both sets:

$$\ell_t = \begin{bmatrix} W_{\text{text}} h_t \\ P(I) h_t \end{bmatrix}, \qquad p_\theta(y_t \mid \mathcal{C}, y_{<t}) = \frac{\exp(\ell_{t,y_t})}{\sum_{u \in \mathcal{V}_{\text{text}} \cup \mathcal{V}_{\text{vis}}(I)} \exp(\ell_{t,u})}. \tag{1}$$

The context $\mathcal{C}$ contains the image and task inputs, and $y_{<t}$ is the generated prefix. Text and references are normalized within the same distribution, allowing both to appear in one autoregressive completion. Only references belonging to the current observation are available to that example.

When a reference v_j is generated, its image-dependent prototype p_j becomes the input embedding at that token position for subsequent decoding. Text tokens retain their learned embeddings.

3.4 From Visual References to Gaze Prediction

Gaze prediction requires a dense heatmap, so we equip the visual-reference representation with a task-specific gaze adapter. At the position $t^\star$ preceding the reference token, the VLM's final-layer hidden state $h_{t^\star}$ is matched against the image-dependent prototypes. The resulting scores retain the spatial correspondence of the patch grid:

$$z_{t^\star} = P(I) h_{t^\star} \in \mathbb{R}^N, \qquad \hat{G} = f_\phi(z_{t^\star}). \tag{2}$$

The adapter f_ϕ normalizes these scores across patches, reshapes them onto the spatial grid, and decodes the dense map through five convolutional layers with bilinear upsampling. We train the VLM and adapter jointly: annotated attention text and spatial references supervise the completion through a language-modeling loss, while recorded gaze maps supervise the heatmap output through binary cross-entropy and KL divergence.

At inference, the image, scene description, and task instruction are followed by a fixed prefix ending at $t^\star$. A single forward pass then produces the heatmap through Eq. 2, without supplying the ground-truth reference or attention explanation. The adapter architecture and training objective are detailed in Appendix A.2.1, and the prompt templates in Appendix A.2.2.

3.5 From Visual Focus to End-to-End Planning

FocusDrive expresses visual focus within the sequence that generates the driving plan. When a focus span is generated, its object description and patch reference provide spatial grounding and image-conditioned visual information for the subsequent action plan and future waypoints.

3.5.1 Supervised Learning of Focus-Based Reasoning

To teach the planner this progression, we organize an annotated completion Y as

$$\underbrace{[d_o,\, v_j]}_{\text{visual focus}} \longrightarrow \underbrace{a}_{\text{action plan}} \longrightarrow \underbrace{w}_{\text{waypoint text}}. \tag{3}$$

Here, d_o is the description of the selected object and v_j references its image patch. The action plan a expresses the intended driving behavior, and w represents the future trajectory τ as numerical text and delimiters encoded by the ordinary text tokenizer. Thus, a and w are both text-token sequences, whereas v_j is an image-dependent visual reference. These spans use `<focus>`, `<plan>`, and `<answer>` delimiters, respectively. See Appendix A.3.2 for input and completion formats.

The pretrained VLM already supports language generation, whereas image-patch references introduce a new vocabulary for selecting visual focus. We begin with visual-reference token (VRT) warm-up, training the model to jointly describe and reference objects before learning complete planning sequences. Before selective driving-focus screening, we construct broader object-reference annotations across Navtrain, describing and localizing relevant objects in each annotated scene. Each warm-up completion contains multiple object descriptions, including relative position, and their corresponding patch references, without action-plan or waypoint targets.

Both reference warm-up and planning SFT use token-level cross-entropy, with stage-specific contexts and target completions:

$$\mathcal{L}_{\text{CE}}(\theta; \mathcal{D}) = -\mathbb{E}_{(\mathcal{C},Y)\sim\mathcal{D}} \sum_{t=1}^{|Y|} \log p_\theta(y_t \mid \mathcal{C}, y_{<t}). \tag{4}$$

Here, $\mathcal{C}$ contains the image and task inputs. During warm-up, $\mathcal{D} = \mathcal{D}_{\text{vrt}}$ and the target contains object descriptions and patch references. During planning SFT, $\mathcal{D} = \mathcal{D}_{\text{plan}}$ and the target contains the focus span, when annotated, followed by the action plan and waypoint text. Scenes without focus annotations still supervise the plan and future motion. Appendices A.3.2 and A.3.3 specify the completion formats and stage-specific training configurations.

3.5.2 Reinforcement Fine-Tuning with GRPO

Supervised learning establishes how to express visual focus and continue from it to a driving plan. Reinforcement fine-tuning adds feedback on the resulting behavior, allowing the policy to improve beyond matching the annotated completion. A driving scene can admit several plausible plans, whose quality is not fully captured by matching a single annotated sequence. Following outcome-based policy optimization (Zhou et al., 2025b; Shang et al., 2026; Li et al., 2026b), we use group relative policy optimization (GRPO) to compare sampled completions for the same observation and reinforce those with better driving outcomes.

Let π_θ denote the VLM policy used for planning. For each observation x, the current policy samples a group of G completions $\{Y_i\}_{i=1}^{G}$. The waypoint text in each completion is parsed into a candidate trajectory τ_i, and the completion receives the PDMS of this trajectory as its reward, $r_i = \text{PDMS}(\tau_i; x)$. This reward evaluates collision avoidance, road compliance, time to collision, comfort, and progress, connecting the generated reasoning sequence to the quality of its resulting driving behavior. Comparing candidates for the same scene gives a relative advantage

$$A_i = \frac{r_i - \bar{r}}{\sigma_r + \delta}, \qquad \bar{r} = \frac{1}{G}\sum_{i=1}^{G} r_i, \tag{5}$$

where σ_r is the within-group reward standard deviation and δ is a numerical stabilizer.

The group-relative loss is

$$\mathcal{L}_{\text{RL}}(\theta) = -\frac{1}{G}\sum_{i=1}^{G}\frac{A_i}{|Y_i|}\sum_{t=1}^{|Y_i|}\log \pi_\theta(y_{i,t} \mid x, y_{i,<t}) + \beta\widehat{\mathcal{K}}(\pi_\theta, \pi_{\text{ref}}). \tag{6}$$

Here, π_{ref} is the frozen SFT policy and β controls KL regularization. The sampled completions and their advantages are treated as fixed during the policy update. The loss covers the generated reasoning and waypoint tokens, while KL regularization constrains deviation from the SFT reference. Appendix A.3.3 provides the KL estimator, token masking, and optimization settings.

4 Experiments

Our experiments examine the progression from modeling visual focus to reasoning with it. Gaze prediction evaluates the spatial accuracy of the focus representation, while planning evaluates its value for action generation. Ablations examine explicit focus supervision, visual-reference preparation, and the use of selected visual features during planning.

4.1 Experimental Setup

Driver gaze benchmark. W3DA (Zhou et al., 2025a) provides language-enriched driver gaze data spanning Normal (DR(eye)VE (Alletto et al., 2016) and LBW (Kasahara et al., 2022)), Safety (BDD-A (Xia et al., 2018)), and Accident (DADA-2000 (Fang et al., 2019)) scenarios. We follow the G-W3DA data preparation (Ke et al., 2026), using 39,673 training and 21,985 test frames. Gaze maps are evaluated at $H_g \times W_g = 256 \times 256$. We report KL divergence (KLdiv), correlation (CC), similarity (SIM), and AUC-Judd/AUC-Borji (AUC-J/AUC-B), with Normal results averaged over all DR(eye)VE and LBW test frames.

Planning benchmark. NAVSIM (Dauner et al., 2024) evaluates end-to-end planning through non-reactive simulation on recorded driving scenes. Following prior driving VLAs (Luo et al., 2025; Yang et al., 2026), we train on Navtrain and evaluate on the 12,146 Navtest scenarios using both PDMS from NAVSIM v1 and EPDMS introduced in NAVSIM v2. The two metrics assess complementary aspects of planning safety, compliance, comfort, and progress.

Table 1: Driver attention prediction across Normal, Safety, and Accident scenarios. Bold and underlined values indicate the best and second-best results. † denotes joint attention and language modeling.

	Normal					Safety					Accident				
Method	KLdiv↓	CC↑	SIM↑	AUC-J↑	AUC-B↑	KLdiv↓	CC↑	SIM↑	AUC-J↑	AUC-B↑	KLdiv↓	CC↑	SIM↑	AUC-J↑	AUC-B↑
Conventional attention predictors															
MLNet (Dodge & Karam, 2018)	2.129	0.547	0.460	0.914	0.836	1.953	0.528	0.433	0.928	0.874	2.897	0.344	0.288	0.893	0.784
ConvNeXt (Liu et al., 2022)	2.042	0.570	0.412	0.916	0.848	1.765	0.567	0.413	0.938	0.877	3.049	0.377	0.248	0.891	0.806
ERFNet (Romera et al., 2017)	1.979	0.558	0.425	0.923	0.840	1.593	0.538	0.410	0.942	0.868	2.181	0.391	0.253	0.930	0.846
Vision–language methods															
GazeXplain† (Chen et al., 2025)	2.578	0.477	0.389	0.857	0.866	2.769	0.383	0.321	0.848	0.743	3.109	0.371	0.236	0.902	0.804
LLada† (Zhou et al., 2025a)	1.219	0.583	0.436	0.952	0.908	1.230	0.579	0.420	0.950	0.912	1.927	0.396	0.262	0.934	0.889
FSDAM† (Hamid et al., 2026)	1.408	0.577	0.467	0.959	0.862	1.222	0.589	0.467	0.958	0.890	1.838	0.439	0.323	0.949	0.872
DualGaze-VLM† (Ke et al., 2026)	1.229	0.613	0.468	0.964	0.900	1.106	0.618	**0.550**	0.961	0.918	1.700	0.471	0.325	0.952	0.898
FocusDrive (Ours)†	**1.137**	**0.639**	**0.491**	**0.969**	**0.917**	**0.959**	**0.670**	0.511	**0.971**	**0.937**	**1.561**	**0.515**	**0.357**	**0.961**	**0.922**

Implementation details. FocusDrive uses Qwen3-VL-2B (Bai et al., 2025) with a visual-reference projector of bottleneck width $r = 64$. We train the gaze model on 2 GPUs and the planner on 16 GPUs with 96 GB of memory each. For the reported gaze experiments, we train for three epochs with a per-GPU batch size of 12, using learning rates of 10^{-5} (VLM) and 10^{-4} (adapter).

For planning, the input comprises a 1920×1080 front-camera image, navigation command, and ego velocity and acceleration. We add focus annotations to 13.1% of the existing Navtrain scenes and retain planning supervision on the full Navtrain set. Appendix A.3.3 provides the stage-specific optimization, rollout, and reward settings.

4.2 Main Results

Driver gaze prediction. Table 1 shows strong performance across Normal, Safety, and Accident scenarios, with FocusDrive leading most reported metrics. It achieves the strongest KLdiv, CC, and AUC results across all three scenario groups, together with the highest SIM in Normal and Accident scenes. The largest correlation gain over DualGaze-VLM occurs in safety-critical scenes, supporting the representation's ability to localize driver focus in demanding situations. These results provide a strong spatial modeling foundation for studying visual focus in planning reasoning.

End-to-end planning. FocusDrive reaches 91.1 PDMS and 91.1 EPDMS (Tables 2 and 3), leading the explicit CoT planners compared here. The comparison includes text-based reasoning, adaptive thinking, exploration-oriented training, and spatial prompting (Luo et al., 2025; Chen et al., 2026; Yang et al., 2026), as well as perception–prediction–planning chains (Ye et al., 2026). These results support organizing driving reasoning around an explicit connection between the relevant object, its image location, and the intended response. FocusDrive turns this connection into a multimodal reasoning sequence that guides end-to-end planning. Strong collision avoidance and road compliance support the PDMS result. FocusDrive leads NC and TTC and ties for the highest TLC in Table 3, aligning with the annotation emphasis on vehicle interactions and traffic-control cues. Together with the gaze results, these findings support the progression from modeling where drivers attend to using visual focus in end-to-end planning.

4.3 Ablation Studies and Analysis

Structured reasoning through visual focus. Table 4 compares FocusDrive with our text CoT baseline, which follows the language-based reasoning format of Curious-VLA before waypoint generation (Chen et al., 2026). FocusDrive adds focus supervision and structures the completion around an object description and image-patch reference, followed by the driving plan. This complete formulation improves SFT performance by 1.3 PDMS and 1.4 EPDMS. After reinforcement fine-tuning, the gains remain 0.9 and 1.0 points, respectively, supporting focus-based reasoning as a stronger basis for planning at both stages.

Focus representation and reference preparation. Table 5 compares focus representations under SFT. Text-2D expresses the focal point using ordinary text coordinates, whereas FocusDrive uses patch references. Both receive planning-focus supervision on the same 13.1% of Navtrain and planning supervision on the full training set. Both improve over the text CoT baseline, supporting the

Table 2: Planning results on NAVSIM Navtest measured by PDMS. Bold and underlined values denote the best and second-best results in each column.

Method	NC↑	DAC↑	TTC↑	Comf.↑	EP↑	PDMS↑
End-to-end planners						
TransFuser (Chitta et al., 2023)	97.7	92.8	92.8	**100.0**	79.2	84.0
DiffusionDrive (Liao et al., 2025)	98.2	96.2	94.7	**100.0**	82.2	88.1
WoTE (Li et al., 2025)	98.5	96.8	94.9	99.9	81.9	88.3
DriveDPO (Shang et al., 2025)	98.5	98.1	94.8	99.9	84.3	90.0
Vision-language-action models						
FSDrive (Zeng et al., 2025)	98.2	93.8	93.3	99.9	80.1	85.1
AutoVLA (Zhou et al., 2025b)	98.4	95.6	**98.0**	99.9	81.9	89.1
DriveVLA-W0 (Li et al., 2026a)	98.7	**99.1**	95.3	99.3	83.3	90.2
AdaThinkDrive (Luo et al., 2025)	98.4	97.8	95.2	**100.0**	84.4	90.3
Curious-VLA (Chen et al., 2026)	98.4	96.9	97.9	98.1	**88.5**	90.3
DriveTeach-VLA (Yang et al., 2026)	98.5	96.9	97.9	98.2	**88.5**	90.4
AutoDrive-P^3 (Ye et al., 2026)	99.1	97.4	96.5	**100.0**	84.8	90.6
FocusDrive (SFT)	98.7	96.8	95.5	**100.0**	82.4	88.9
FocusDrive (RL)	**99.3**	98.8	97.7	**100.0**	82.8	**91.1**

Table 3: Planning results on the same NAVSIM Navtest split measured by EPDMS. Bold and underlined values denote the best and second-best results in each column.

Method	NC↑	DAC↑	DDC↑	TLC↑	EP↑	TTC↑	LK↑	HC↑	EC↑	EPDMS↑
End-to-end planners										
TransFuser (Chitta et al., 2023)	96.9	89.9	97.8	99.7	87.1	95.4	92.7	98.3	87.2	84.0
WoTE (Li et al., 2025)	98.5	96.8	98.8	99.8	86.1	97.9	95.5	98.3	82.9	87.7
DiffusionDrive (Liao et al., 2025)	98.2	96.2	99.5	99.8	87.4	97.3	96.9	**98.4**	**87.7**	88.2
Vision-language-action models										
DriveVLA-W0 (Li et al., 2026a)	98.5	**99.1**	98.0	99.7	86.4	98.1	93.2	97.9	58.9	86.1
DriveWorld-VLA (jia et al., 2026)	98.6	**99.1**	**99.6**	99.8	87.4	97.9	97.0	97.8	78.6	86.8
Latent-WAM (Wang et al., 2026)	98.1	97.3	**99.6**	99.8	87.7	97.3	97.6	98.1	87.3	89.3
DriveFuture (Hong et al., 2026)	99.1	97.2	99.3	99.8	**87.8**	98.5	96.1	98.3	85.2	89.9
FocusDrive (SFT)	98.7	96.8	99.5	**99.9**	86.7	98.0	**97.8**	98.3	87.5	89.1
FocusDrive (RL)	**99.3**	98.8	**99.6**	**99.9**	84.9	**98.9**	97.1	**98.4**	87.2	**91.1**

use of spatially grounded VLM reasoning for driving planning. Patch references exceed Text-2D by 0.4 points on both metrics without reference warm-up and by 0.6 points with warm-up. Reference warm-up further improves FocusDrive by 0.3 PDMS and 0.2 EPDMS.

Image-conditioned patch references. To examine the reference input, we replace the selected patch prototype with a learned reference-ID embedding while retaining dynamic patch selection. Both variants retain spatial indexing. A learned ID embedding encodes the grid location with a vector shared across images, whereas a patch prototype combines that spatial correspondence with image-conditioned visual semantics. Given the spatial information already available through the reference ID, using the corresponding patch prototype additionally supplies a visual representation of the selected region in the current image. This image-conditioned input improves PDMS and EPDMS by a further 0.3 points each, indicating a benefit beyond the location information retained by the ID variant. Patch references thus serve both to express the selected focus and to carry its visual representation into subsequent reasoning. Combined with reference warm-up, this formulation achieves the highest scores in Table 5. Together with the larger gains over the text CoT baseline, these comparisons support the complete focus-based formulation: explicit spatial focus improves planning, and the configuration that also incorporates image-conditioned reference inputs performs best among the variants evaluated under SFT.

Qualitative analysis. Figure 3 examines how selected objects, generated reasoning, and driving responses correspond within individual scenes. FocusDrive's generated focus and plan are shown alongside the predicted trajectories of the text CoT baseline and FocusDrive, while patch references identify the image regions explicitly selected by FocusDrive. These examples illustrate how spatial

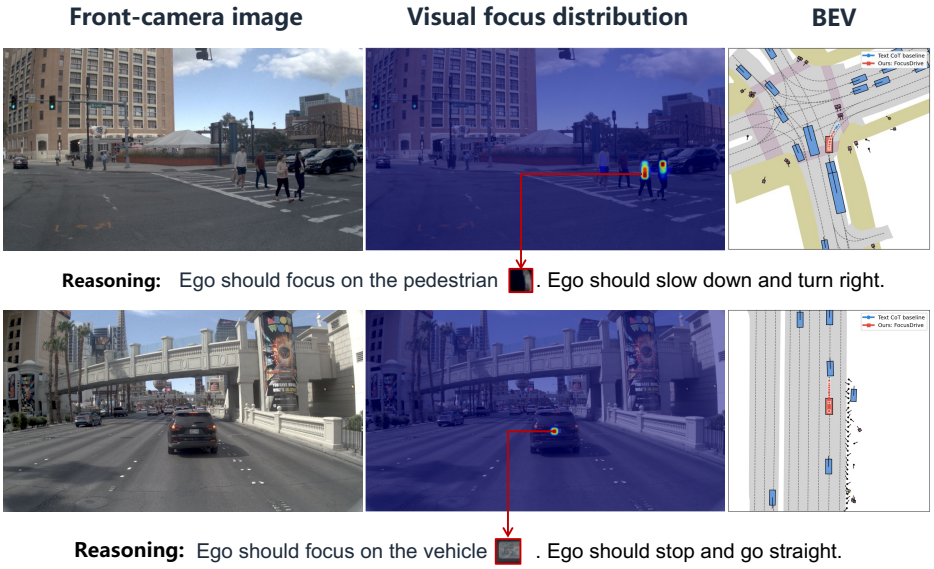

Figure 3: **Visual focus for planning.** Scene images, visual focus distributions, and predicted trajectories with FocusDrive's reasoning. The text CoT baseline (blue) receives an NC score of 0 in the top case and a TTC score of 0 in the bottom case; FocusDrive (red) avoids the corresponding penalties in both cases.

Table 4: Comparison with our text CoT baseline under SFT and reinforcement fine-tuning. Δ denotes the gain over the baseline in the corresponding metric and training stage.

Method	PDMS↑	Δ	EPDMS↑	Δ
Supervised fine-tuning				
Text CoT baseline	87.6	–	87.7	–
FocusDrive	**88.9**	**+1.3**	**89.1**	**+1.4**
Reinforcement fine-tuning				
Text CoT baseline	90.2	–	90.1	–
FocusDrive	**91.1**	**+0.9**	**91.1**	**+1.0**

Table 5: Focus representations and components under SFT. Ref. input: coordinates (Text), learned ID embeddings (ID), or image-conditioned patch prototypes (Patch ref.).

Method	Warm-up	Ref. input	PDMS↑	EPDMS↑
Text-2D	–	Text	88.2	88.5
FocusDrive	–	Patch ref.	88.6	88.9
Text-2D	✓	Text	88.3	88.5
FocusDrive	✓	ID	88.6	88.8
FocusDrive	✓	Patch ref.	**88.9**	**89.1**

focus is expressed within reasoning and how the resulting plans differ. Appendices A.2.3 and A.3.4 provide gaze comparisons and further planning visualizations, respectively.

5 Conclusion

We presented FocusDrive, a framework that makes visual focus an explicit part of driving reasoning. By pairing object descriptions with generated patch references, it connects decision-relevant objects to their visual representations within the planning sequence. Strong gaze-prediction results provide a foundation for modeling focus, while NAVSIM experiments demonstrate the benefits of incorporating explicit focus into driving reasoning. FocusDrive thus connects the study of where to focus with the practical question of how to act on what is seen.

AI use statement

We used generative AI tools to assist with code development, literature and reference checking, manuscript drafting and revision, and critical discussion of the methodology and interpretation of experimental results. We also used Qwen3.7-Plus to propose candidate objects during offline driving-focus annotation, as detailed in Appendix A.3.1. These proposals were reviewed by human

annotators, who selected image locations, revised object descriptions, and removed incorrect annotations. AI-assisted manuscript revisions were reviewed and revised by the authors. The authors take full responsibility for the methods, results, claims, and final content of this paper.

Reproducibility statement

Sections 3 and 4.1 describe the model, annotation procedure, training setup, and evaluation metrics. Appendix A provides supplementary implementation details to support reproduction of the experiments.

References

Stefano Alletto, Andrea Palazzi, Francesco Solera, Simone Calderara, and Rita Cucchiara. Dr (eye) ve: a dataset for attention-based tasks with applications to autonomous and assisted driving. In *Proceedings of the ieee conference on computer vision and pattern recognition workshops*, pp. 54–60, 2016.

Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, Wenbin Ge, Zhifang Guo, Qidong Huang, Jie Huang, Fei Huang, Binyuan Hui, Shutong Jiang, Zhaohai Li, Mingsheng Li, Mei Li, Kaixin Li, Zicheng Lin, Junyang Lin, Xuejing Liu, Jiawei Liu, Chenglong Liu, Yang Liu, Dayiheng Liu, Shixuan Liu, Dunjie Lu, Ruilin Luo, Chenxu Lv, Rui Men, Lingchen Meng, Xuancheng Ren, Xingzhang Ren, Sibo Song, Yuchong Sun, Jun Tang, Jianhong Tu, Jianqiang Wan, Peng Wang, Pengfei Wang, Qiuyue Wang, Yuxuan Wang, Tianbao Xie, Yiheng Xu, Haiyang Xu, Jin Xu, Zhibo Yang, Mingkun Yang, Jianxin Yang, An Yang, Bowen Yu, Fei Zhang, Hang Zhang, Xi Zhang, Bo Zheng, Humen Zhong, Jingren Zhou, Fan Zhou, Jing Zhou, Yuanzhi Zhu, and Ke Zhu. Qwen3-vl technical report, 2025. URL https://arxiv.org/abs/2511.21631.

Canyu Chen, Yuguang Yang, Zhewen Tan, Yizhi Wang, Ruiyi Zhan, Haiyan Liu, Xuanyao Mao, Jason Bao, Xinyue Tang, Linlin Yang, Bingchuan Sun, Yan Wang, and Baochang Zhang. Devil is in narrow policy: Unleashing exploration in driving vla models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Findings*, pp. 1062–1072, June 2026.

Xianyu Chen, Ming Jiang, and Qi Zhao. Gazexplain: Learning to predict natural language explanations of visual scanpaths. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 314–333, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-73242-3.

Hao Cheng, Yanbo Jiang, Wenhao Yu, Rui Zhou, Jiang Bian, Keyu Chen, Zhiyuan Liu, Heye Huang, Hailun Zhang, Fang Zhang, Jianqiang Wang, and Sifa Zheng. Driving risk emerges from the required two-dimensional joint evasive acceleration, 2026. URL https://arxiv.org/abs/2604.17841.

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. doi: 10.1109/TPAMI.2022.3200245.

Marcella Cornia, Lorenzo Baraldi, Giuseppe Serra, and Rita Cucchiara. A deep multi-level network for saliency prediction. In *2016 23rd International Conference on Pattern Recognition (ICPR)*, pp. 3488–3493, 2016. doi: 10.1109/ICPR.2016.7900174.

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 A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang (eds.), *Advances in Neural Information Processing Systems*, volume 37, pp. 28706–28719. Curran Associates, Inc., 2024. doi: 10.52202/079017-0902. URL https://proceedings.neurips.cc/paper_files/paper/2024/file/

32768f7faf1995026ef9821c696f3404-Paper-Datasets_and_Benchmarks_Track.pdf.

Samuel F Dodge and Lina J Karam. Visual saliency prediction using a mixture of deep neural networks. *IEEE Transactions on Image Processing*, 27(8):4080–4090, 2018.

Jianwu Fang, Dingxin Yan, Jiahuan Qiao, Jianru Xue, He Wang, and Sen Li. Dada-2000: Can driving accident be predicted by driver attention? analyzed by a benchmark. In *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, pp. 4303–4309. IEEE, 2019.

Jianwu Fang, Dingxin Yan, Jiahuan Qiao, Jianru Xue, and Hongkai Yu. Dada: Driver attention prediction in driving accident scenarios. *IEEE Transactions on Intelligent Transportation Systems*, 23(6):4959–4971, 2022. doi: 10.1109/TITS.2020.3044678.

Haoyu Fu, Diankun Zhang, Zongchuang Zhao, Jianfeng Cui, Dingkang Liang, Chong Zhang, Dingyuan Zhang, Hongwei Xie, Bing Wang, and Xiang Bai. Orion: A holistic end-to-end autonomous driving framework by vision-language instructed action generation. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 24823–24834, 2025. doi: 10.1109/ICCV51701.2025.02302.

Kaiser Hamid, Can Cui, Khandakar Ashrafi Akbar, Ziran Wang, and Nade Liang. Fsdam: Few-shot driving attention modeling via vision-language coupling, 2026. URL https://arxiv.org/abs/2511.12708.

John C. Hayward. Near-miss determination through use of a scale of danger. *Highway Research Record*, (384):24–34, 1972. URL https://onlinepubs.trb.org/Onlinepubs/hrr/1972/384/384-004.pdf.

Yufeng Hong, Xiaotian Zhou, Yingyan Li, Xiangpo Zhou, Lin Liu, Yadan Luo, Shaoqing Xu, Lei Yang, and Ziying Song. Drivefuture: Future-aware latent world models for autonomous driving, 2026. URL https://arxiv.org/abs/2605.09701.

Zhijian Huang, Tao Tang, Shaoxiang Chen, Sihao Lin, Zequn Jie, Lin Ma, Guangrun Wang, and Xiaodan Liang. Making large language models better planners with reasoning-decision alignment. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 73–90, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-72764-1.

Jyh-Jing Hwang, Runsheng Xu, Hubert Lin, Wei-Chih Hung, Jingwei Ji, Kristy Choi, Di Huang, Tong He, Paul Covington, Benjamin Sapp, Yin Zhou, James Guo, Dragomir Anguelov, and Mingxing Tan. Emma: End-to-end multimodal model for autonomous driving, 2025. URL https://arxiv.org/abs/2410.23262.

Feiyang jia, Lin Liu, Ziying Song, Caiyan Jia, Hangjun Ye, Xiaoshuai Hao, and Long Chen. Driveworld-vla: Unified latent-space world modeling with vision-language-action for autonomous driving, 2026. URL https://arxiv.org/abs/2602.06521.

Isaac Kasahara, Simon Stent, and Hyun Soo Park. Look both ways: Self-supervising driver gaze estimation and road scene saliency. In *Computer Vision – ECCV 2022*, pp. 126–142, Cham, 2022. Springer Nature Switzerland.

Zehong Ke, Yanbo Jiang, Jinhao Li, Zhiyuan Liu, Yiqian Tu, Qingwen Meng, Heye Huang, and Jianqiang Wang. From scene to object: Text-guided dual-gaze prediction. *IEEE Internet of Things Journal*, 2026. doi: 10.1109/JIOT.2026.3721773.

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 *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 27137–27146, 2025. doi: 10.1109/ICCV51701.2025.02519.

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 C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 7890–7911, 2026a. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/0d70423f59c5fdd24f0dd3fa52e34623-Paper-Conference.pdf.

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 C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 157518–157556, 2026b. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/ff7bf6014f7826da531aa50f4538ee19-Paper-Conference.pdf.

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 (CVPR)*, pp. 12037–12047, June 2025.

Zhuang Liu, Hanzi Mao, Chao-Yuan Wu, Christoph Feichtenhofer, Trevor Darrell, and Saining Xie. A convnet for the 2020s. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pp. 11976–11986, 2022.

Jinghui Lu, Jiayi Guan, Zhijian Huang, Jinlong Li, Guang Li, Lingdong Kong, Yingyan Li, Han Wang, Shaoqing Xu, Yuechen Luo, Fang Li, Chenxu Dang, Junli Wang, Tao Xu, Jing Wu, Jianhua Wu, Xiaoshuai Hao, Wen Zhang, Tianyi Jiang, Lingfeng Zhang, Lei Zhou, Yingbo Tang, Jie Wang, Yinfeng Gao, Xizhou Bu, Haochen Tian, Yihang Qiu, Feiyang Jia, Lin Liu, Yigu Ge, Hanbing Li, Yuannan Shen, Jianwei Cui, Hongwei Xie, Bing Wang, Haiyang Sun, Jingwei Zhao, Jiahui Huang, Pei Liu, Zeyu Zhu, Yuncheng Jiang, Zibin Guo, Chuhong Gong, Hanchao Leng, Kun Ma, Naiyan Wang, Guang Chen, Kuiyuan Yang, Hangjun Ye, and Long Chen. Xiaomi onevl: One-step latent reasoning and planning with vision-language explanation, 2026. URL https://arxiv.org/abs/2604.18486.

Yuechen Luo, Fang Li, Shaoqing Xu, Zhiyi Lai, Lei Yang, Qimao Chen, Ziang Luo, Zixun Xie, Shengyin Jiang, Jiaxin Liu, Long Chen, Bing Wang, and Zhi xin Yang. Adathinkdrive: Adaptive thinking via reinforcement learning for autonomous driving, 2025. URL https://arxiv.org/abs/2509.13769.

Alexander Makrigiorgos, Ali Shafti, Alex Harston, Julien Gerard, and A. Aldo Faisal. Human visual attention prediction boosts learning & performance of autonomous driving agents, 2019. URL https://arxiv.org/abs/1909.05003.

Andrea Palazzi, Davide Abati, simone Calderara, Francesco Solera, and Rita Cucchiara. Predicting the driver's focus of attention: The dr(eye)ve project. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 41(7):1720–1733, 2019. doi: 10.1109/TPAMI.2018.2845370.

Yiming Qin, Bomin Wei, Jiaxin Ge, Konstantinos Kallidromitis, Stephanie Fu, Trevor Darrell, and XuDong Wang. Chain-of-visual-thought: Teaching vlms to see and think better with continuous visual tokens, 2026. URL https://arxiv.org/abs/2511.19418.

Eduardo Romera, José M Alvarez, Luis M Bergasa, and Roberto Arroyo. Erfnet: Efficient residual factorized convnet for real-time semantic segmentation. *IEEE Transactions on Intelligent Transportation Systems*, 19(1):263–272, 2017.

Shuyao Shang, Yuntao Chen, Yuqi Wang, Yingyan Li, and ZHAO-XIANG ZHANG. Drivedpo: Policy learning via safety dpo for end-to-end autonomous driving. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 81565–81585. Curran Associates, Inc., 2025. doi: 10.52202/085713-2724. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/750337e1301941f81ae31a90e0a1c181-Paper-Conference.pdf.

Shuyao Shang, Bing Zhan, Yunfei Yan, Yuqi Wang, Yingyan Li, Yasong An, Xiaoman Wang, Jierui Liu, Lu Hou, Lue Fan, Zhaoxiang Zhang, and Tieniu Tan. DynVLA: Learning world dynamics for action reasoning in autonomous driving. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=UB6M4BSvwJ.

Chonghao Sima, Katrin Renz, Kashyap Chitta, Li Chen, Hanxue Zhang, Chengen Xie, Jens Beißwenger, Ping Luo, Andreas Geiger, and Hongyang Li. Drivelm: Driving with graph visual question answering. In Aleš Leonardis, Elisa Ricci, Stefan Roth, Olga Russakovsky, Torsten Sattler, and Gül Varol (eds.), *Computer Vision – ECCV 2024*, pp. 256–274, Cham, 2025. Springer Nature Switzerland. ISBN 978-3-031-72943-0.

Tianhui Song, Haoyu Lu, Hao Yang, Lin Sui, Haoning Wu, Zaida Zhou, Zhiqi Huang, Yiping Bao, Y. Charles, Xinyu Zhou, and Limin Wang. Towards pixel-level vlm perception via simple points prediction, 2026. URL https://arxiv.org/abs/2601.19228.

Yongyi Su, Haojie Zhang, Shijie Li, Nanqing Liu, Jingyi Liao, Junyi Pan, Yuan Liu, Xiaofen Xing, Chong Sun, Chen Li, Nancy Chen, Shuicheng YAN, xulei yang, and Xun Xu. Patch-as-decodable-token: Towards unified multi-modal vision tasks in mllms. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 147893–147919, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/ef3644ca8787131263600445b0461383-Paper-Conference.pdf.

Xiaoyu Tian, Junru Gu, Bailin Li, Yicheng Liu, Yang Wang, Zhiyong Zhao, Kun Zhan, Peng Jia, XianPeng Lang, and Hang Zhao. Drivevlm: The convergence of autonomous driving and large vision-language models. In Pulkit Agrawal, Oliver Kroemer, and Wolfram Burgard (eds.), *Proceedings of The 8th Conference on Robot Learning*, volume 270 of *Proceedings of Machine Learning Research*, pp. 4698–4726. PMLR, 06–09 Nov 2025. URL https://proceedings.mlr.press/v270/tian25c.html.

Jianqiang Wang, Jian Wu, and Yang Li. The driving safety field based on driver–vehicle–road interactions. *IEEE Transactions on Intelligent Transportation Systems*, 16(4):2203–2214, 2015. doi: 10.1109/TITS.2015.2401837.

Linbo Wang, Yupeng Zheng, Qiang Chen, Shiwei Li, Yichen Zhang, Zebin Xing, Qichao Zhang, Xiang Li, Deheng Qian, Pengxuan Yang, Yihang Dong, Ce Hao, Xiaoqing Ye, Junyu han, Yifeng Pan, and Dongbin Zhao. Latent-wam: Latent world action modeling for end-to-end autonomous driving, 2026. URL https://arxiv.org/abs/2603.24581.

Ye Xia, Danqing Zhang, Jinkyu Kim, Ken Nakayama, Karl Zipser, and David Whitney. Predicting driver attention in critical situations. In *Asian conference on computer vision*, pp. 658–674. Springer, 2018.

Yuguang Yang, Canyu Chen, Zhewen Tan, Yizhi Wang, Zichao Feng, Chunyang Liu, Kehua Sheng, Juan Zhang, Linlin Yang, Baochang Zhang, Yan Wang, Bo Zhang, and Xianbin Cao. Teaching vision-language-action models what to see and where to look, 2026. URL https://arxiv.org/abs/2607.01658.

Yuqi Ye, Zijian Zhang, Junhong Lin, Shangkun Sun, Changhao Peng, and Wei Gao. Autodrive-p3: Unified chain of perception-prediction-planning thought via reinforcement fine-tuning. In C. Vondrick, B. Hariharan, C. Raffel, L. Pinto, D. Yang, and A. Faust (eds.), *International Conference on Learning Representations*, volume 2026, pp. 31007–31036, 2026. URL https://proceedings.iclr.cc/paper_files/paper/2026/file/34515e20a0ec8c894c29fbd65d403678-Paper-Conference.pdf.

Shuang Zeng, Xinyuan Chang, Mengwei Xie, Xinran Liu, Yifan Bai, Zheng Pan, Mu Xu, and Xing Wei. Futuresightdrive: Thinking visually with spatio-temporal cot for autonomous driving. In D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 67299–67318. Curran Associates, Inc., 2025. doi: 10.52202/085713-2257. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/61466f2c7a87edfa5898665c70af0e90-Paper-Conference.pdf.

Qingqing Zhao, Yao Lu, Moo Jin Kim, Zipeng Fu, Zhuoyang Zhang, Yecheng Wu, Zhaoshuo Li, Qianli Ma, Song Han, Chelsea Finn, Ankur Handa, Tsung-Yi Lin, Gordon Wetzstein, Ming-Yu Liu, and Donglai Xiang. Cot-vla: Visual chain-of-thought reasoning for vision-language-action models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1702–1713, June 2025.

Yuchen Zhou, Jiayu Tang, Xiaoyan Xiao, Yueyao Lin, Linkai Liu, Zipeng Guo, Hao Fei, Xiaobo Xia, and Chao Gou. Where, what, why: Towards explainable driver attention prediction. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 2675–2685, 2025a. doi: 10.1109/ICCV51701.2025.00257.

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 D. Belgrave, C. Zhang, H. Lin, R. Pascanu, P. Koniusz, M. Ghassemi, and N. Chen (eds.), *Advances in Neural Information Processing Systems*, volume 38, Main Conference, pp. 27920–27956. Curran Associates, Inc., 2025b. doi: 10.52202/085713-0942. URL https://proceedings.neurips.cc/paper_files/paper/2025/file/2843fccca5bedd369a4764848b9bd546-Paper-Conference.pdf.

A Appendix

A.1 Visual Reference Implementation

Reference prototypes. The visual backbone supplies spatial embeddings in the language model's hidden space. The projector in Sec. 3.3 uses layer normalization followed by a low-rank residual map:

$$\bar{e}_j = \mathrm{LN}(e_j), \qquad p_j = \bar{e}_j + W_{\mathrm{up}} W_{\mathrm{down}} \bar{e}_j, \tag{7}$$

where $W_{\mathrm{down}} \in \mathbb{R}^{r \times d}$ and $W_{\mathrm{up}} \in \mathbb{R}^{d \times r}$ are bias-free linear layers, with $r = 64$. The current image's prototypes supply both the visual-reference output choices and the input embeddings of the generated reference tokens. Ordinary language tokens retain the VLM's text embeddings.

Mapping locations to patches. Each annotated location is assigned to the cell that contains it on the processed image's patch grid. For cell or box annotations, we use the normalized center as the location. Cells are ordered from left to right and top to bottom, so each reference identifies the corresponding visual embedding. The grid is 34×60 for planning and 18×32 for gaze prediction.

A.2 Gaze Prediction

A.2.1 Model and Training Configuration

Inputs. Each training and test frame is resized to 1024×576 (bilinear), which yields a spatial reference grid of $H_p \times W_p = 18 \times 32$, i.e., $N = 576$ reference units. The scenario description is derived from the W3DA question field by truncating it before its question–answer template, so that only the scene description remains and no question or answer text is exposed. Together with the task instruction of App. A.2.2, this forms the task prompt q.

Training. Training pairs each frame with the stage-2 language and bounding-box annotations of the G-W3DA preparation. The completion follows the gaze format of App. A.2.2: a fixed focus template containing one reference v_j, followed by the annotated attention explanation. The reference index is sampled uniformly from the grid cells whose centers fall inside the annotated box. Training minimizes the language-modeling loss on the completion together with a heatmap loss on the decoded map,

$$\mathcal{L} = \mathcal{L}_{\mathrm{LM}} + \lambda_g \mathcal{L}_{\mathrm{gaze}}, \qquad \mathcal{L}_{\mathrm{gaze}} = \mathrm{BCE}\big(\hat{G}, G\big) + \lambda_{\mathrm{KL}} \mathrm{KL}\big(G \,\|\, \hat{G}\big), \tag{8}$$

where BCE denotes binary cross-entropy and both maps are normalized to sum to one for the KL term. We set $\lambda_g = 3$ and $\lambda_{\mathrm{KL}} = 0.5$. We train the gaze model used in evaluation for three epochs with AdamW on 2 GPUs.

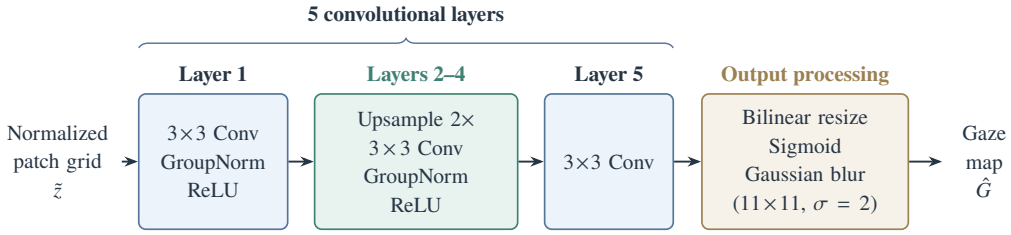

Figure 4: Gaze readout architecture. The middle block is repeated three times, giving five convolutional layers in total. Operations within each block run from top to bottom; upsampling uses bilinear interpolation.

Gaze adapter. We obtain $h_{t^\star}$ from the VLM's final layer at the last token of the fixed prefix in App. A.2.2, which ends with `object` followed by a space, immediately before the visual-reference token. Multiplying this state by the prototype matrix gives the reference scores $z_{t^\star} = P(I)h_{t^\star}$. The adapter applies z-score normalization across patches:

$$\tilde{z}_j = \frac{z_{t^\star,j} - \mu_z}{s_z + 10^{-6}}, \qquad \mu_z = \frac{1}{N}\sum_{j=1}^{N} z_{t^\star,j}, \qquad s_z = \sqrt{\frac{1}{N-1}\sum_{j=1}^{N}(z_{t^\star,j} - \mu_z)^2}. \tag{9}$$

The normalized scores are reshaped onto the patch grid and decoded into a gaze map by the five-layer convolutional readout in Fig. 4.

Evaluation and inference. We report KL divergence (KLdiv), Pearson correlation (CC), similarity (SIM), and AUC-Judd/AUC-Borji (AUC-J/AUC-B). Ground-truth gaze maps are converted to grayscale, resized to 256×256, and scaled to $[0, 1]$. Let i index pixels, $p_i = \hat{G}_i / \sum_j \hat{G}_j$, and $q_i = G_i / \sum_j G_j$. KLdiv measures distributional discrepancy, while CC measures linear correlation:

$$\mathrm{KLdiv} = \sum_i q_i \log \frac{q_i}{p_i}, \qquad \mathrm{CC} = \frac{\sum_i (\hat{G}_i - \bar{\hat{G}})(G_i - \bar{G})}{\sqrt{\sum_i (\hat{G}_i - \bar{\hat{G}})^2 \sum_i (G_i - \bar{G})^2}}, \tag{10}$$

where bars denote pixel means. SIM measures the intersection of two unit-sum distributions (Ke et al., 2026). Let $M(X) = (X - \min X)/(\max X - \min X)$ denote min–max normalization. With this preprocessing, the normalized-distribution definition is

$$\mathrm{SIM} = \sum_i \min\left(\frac{M(\hat{G})_i}{\sum_j M(\hat{G})_j}, \frac{M(G)_i}{\sum_j M(G)_j} \right). \tag{11}$$

Numerical stabilizers are omitted from the metric formulas for clarity.

For AUC, fixation pixels satisfy $M(G)_i \geq 0.7$. Thresholding $M(\hat{G})$ gives the true-positive rate TPR and false-positive rate FPR. AUC-J uses all non-fixation pixels as negatives, whereas AUC-B uses 100 sets of uniformly sampled image pixels as negatives, each matching the number of fixation pixels:

$$\text{AUC-J} = \int \mathrm{TPR}\, d\mathrm{FPR}_{\mathrm{all}}, \qquad \text{AUC-B} = \frac{1}{100}\sum_{b=1}^{100} \int \mathrm{TPR}\, d\mathrm{FPR}_b. \tag{12}$$

Both areas are computed by trapezoidal integration; AUC-J uses prediction values at fixation pixels as thresholds, and AUC-B uses thresholds from 0.1 to 0.9 in steps of 0.1. Metrics are averaged over test frames, with Normal pooling DR(eye)VE and LBW. Given the image and the prompt and prefix in App. A.2.2, a single forward pass produces the predicted heatmap through Eq. 2.

A.2.2 Prompt and Output Formats

The gaze task uses a single user message containing the image and the instruction below, where `<image>` marks the image placeholder and {`scene`} the scenario description of App. A.2.1. This user message is identical in training and evaluation; only the assistant-side continuation differs.

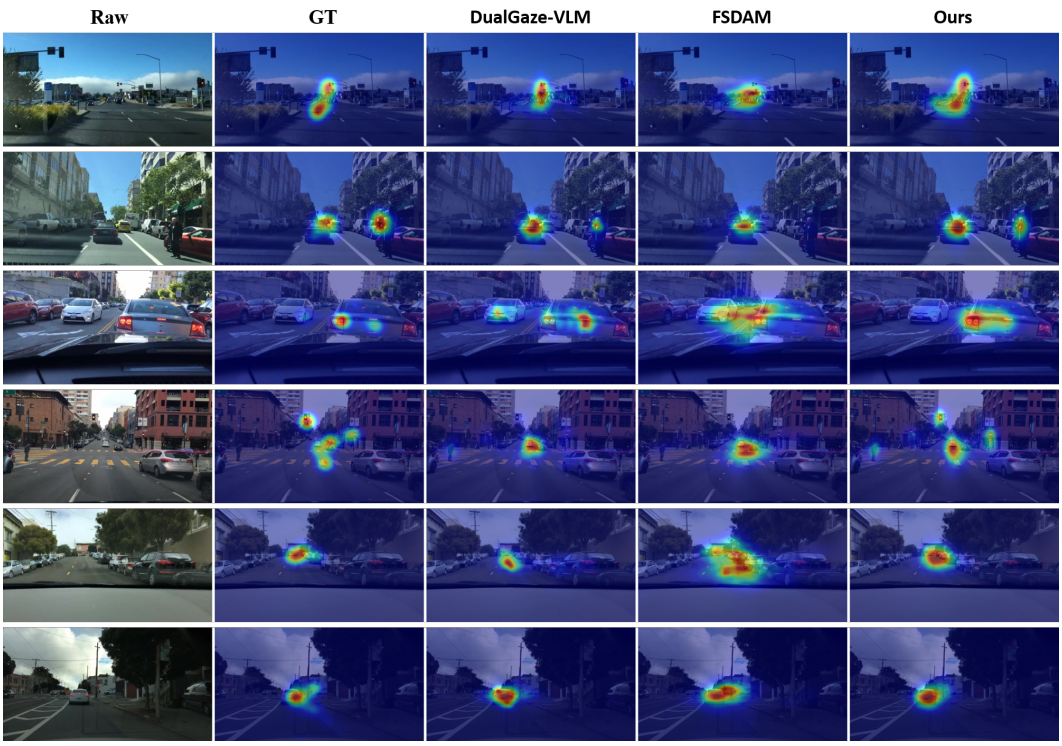

Figure 5: Qualitative gaze comparisons on six safety-critical BDD-A test scenes. Columns from left to right: input scene, ground-truth gaze, DualGaze-VLM, FSDAM, and FocusDrive (ours). FocusDrive predictions are produced by the readout of Eq. 2 without ground-truth focus references or attention explanations.

```
<image> is the front view. {scene} You are the driver. Locate your
primary attention focus: when a safety-critical object needs
attention, output <focus>Ego should focus on the object
<|VRT_n|>.</focus> with exactly one VRT reference, then output the
attention explanation inside <plan></plan>.
```

Training completions place the focus span and the annotated explanation inside a reasoning block delimited by `<think>`; `<|VRT_j|>` is rewritten at runtime to the reference index sampled from the annotated box (App. A.2.1), and {`explanation`} is the annotated attention explanation.

```
<think><focus>Ego should focus on the safety-critical object
<|VRT_j|>.</focus><plan>{explanation}</plan></think>
```

At test time, after the chat template and generation prompt, the following fixed scaffold is appended. It selects the same pre-reference decoding position supervised during training but contains neither a reference index nor any other spatial information. The scaffold ends with a space:

```
<think><focus>Ego should focus on the safety-critical object
```

A.2.3 Qualitative Results

Figure 5 provides example-level evidence complementary to the aggregate results in Table 1. Across the selected scenes, FocusDrive places its dominant responses on the same traffic participants as the ground truth and recovers the separated multi-region structure visible in the first, second, and fourth rows. FSDAM often produces broader or merged responses, as in the first, third, and fifth rows, while DualGaze-VLM weakens or misses secondary regions in several multi-object scenes. The final two rows further show that FocusDrive remains spatially compact in single-target cases. These

Table 6: Object-category distribution among training focus annotations.

Object category	Share (%)
Vehicle	84.43
Traffic light	7.18
Stop sign	5.32
Pedestrian	2.92
Bicycle	0.10
Other	0.04

examples do not replace quantitative evaluation, but illustrate spatial selectivity and the preservation of multiple relevant attention regions without supplying ground-truth focus references or attention explanations.

A.3 End-to-End Planning

A.3.1 Driving-Focus Annotations

Candidate construction. We construct focus annotations within Navtrain through risk-based screening and VLM-assisted candidate selection. Screening combines cues from time to collision, safety-field formulations, and interaction risk assessment (Hayward, 1972; Wang et al., 2015; Cheng et al., 2026). Qwen3.7-Plus complements these cues in traffic-control and yielding situations, using the front image, candidate boxes, and recorded future ego path to propose relevant objects. The recorded future ego path is used only for offline annotation and is not included in the planning policy's observation input during training or inference.

Human selection and consolidation. Two annotators review the proposed objects, manually select their image locations, revise object categories and descriptions, and remove incorrect annotations. The interface supports patch selection on a 34×60 image grid. Each focus-annotated scene currently has a fixed patch target. Future work will involve more annotators to model a distribution over focus locations, capturing variation in human focus selection.

Focus annotations cover 13.1% of Navtrain; the remaining 86.9% of examples supervise the plan and trajectory without a focus span. Among the focus labels, 37.0% come from reviewed risk-based candidates and 63.0% from reviewed traffic-control/yielding candidates. All training examples remain available for planning. We annotate only Navtrain and provide no focus annotations for Navtest, which is reserved for evaluation.

Table 6 shows that vehicles account for 84.43% of the selected focus objects, followed by traffic lights and stop signs. The associated plans involve deceleration in 89.2% of focus-annotated examples. These annotations emphasize collision avoidance and traffic-rule compliance, including potential conflicts and control elements that warrant explicit consideration. They also include scenes flagged for potential risk in which the recorded ego trajectory does not decelerate, so focus selection is not restricted to observed braking. Planning supervision remains available on the full Navtrain set.

A.3.2 Prompt and Output Formats

The user message contains the front-camera image, navigation command, ego velocity and acceleration, with no system turn. The complete planning prompt is reproduced below; braces denote fields filled at runtime.

```
<image> is the front view. Command: {COMMAND}. {ego_state_text}
Before the trajectory, output a required <plan>Ego should ...</plan>
inside <think></think>. When a safety-critical object needs
attention, prepend <focus>Ego should focus on the object
<|VRT_n|>.</focus> with exactly one VRT reference. Do not emit a focus
slot otherwise. Output the predicted trajectory in <answer></answer>.
For the content in <answer></answer>, Only generate the predicted
future waypoints in pure text format:
[x_1, y_1, heading_1], [x_2, y_2, heading_2], ..., [x_8, y_8, heading_8].
Output exactly 8 waypoints in the format [x, y, heading], with numbers
in decimal form and exactly 2 digits after the decimal point.
Separate waypoints using comma. Do not include extra text, brackets,
or invalid values. The output must strictly follow this structure.
```

The assistant places reasoning inside `<think>` and waypoints inside `<answer>`. The schematic completion below illustrates a focus-annotated example; the ellipsis abbreviates the waypoint sequence. Examples without focus annotations omit the focus span and retain the plan and waypoints.

```
<think><focus>Ego should focus on the red traffic light
<|VRT_j|>.</focus><plan>Ego should slow down and go
straight.</plan></think>
<answer>[x_1, y_1, heading_1], ..., [x_8, y_8, heading_8]</answer>
```

The plan expresses speed and directional meta-actions obtained from the open-source release of Curious-VLA (Chen et al., 2026). Text-2D uses the same reviewed focus categories and action plans, expressing the selected location as ordinary text coordinates. The text CoT baseline uses language reasoning followed by the same waypoint targets. In planning SFT, cross-entropy supervises the complete assistant sequence, including visual references; user-prompt tokens are masked. Waypoint supervision uses text-token cross-entropy.

At inference, the model generates whether to include a focus span as part of its completion. Motivated by selective visual focus, we use an optional single reference to highlight a primary decision-relevant object. The reported NAVSIM gains are achieved with this single-focus format, while the model retains access to the full scene. Future work will examine whether multiple explicit references help in more complex interactions.

Learned reference-ID control. The control in Table 5 retains image-dependent output selection: reference logits are computed from the current image's patch prototypes. The selected token's input embedding is instead obtained from a learned table with one vector for each of the 2,040 reference IDs, shared across images. This preserves the reference's spatial index while replacing its image-conditioned input embedding. The model retains the full image input.

A.3.3 Training Configuration

Reference warm-up and supervised fine-tuning. Before selective driving-focus screening, we construct broader object-reference annotations across Navtrain, describing and localizing relevant objects in each annotated scene. Its annotations pair an object description, including relative position, with the corresponding image patch; objects need not be those selected as driving focus for planning. Each warm-up completion jointly generates multiple object descriptions and their corresponding patch references, learning to associate object language with spatial visual content. Two epochs of this warm-up precede three epochs of planning SFT. Planning SFT uses full-parameter AdamW with learning rate 4×10^{-5}, weight decay 0.05, and an effective batch size of 64 (16 GPUs, two examples per GPU, and two gradient-accumulation steps). The schedule uses 100 learning-rate warm-up steps followed by cosine decay.

The warmed Text-2D control starts from the PaDT reference warm-up and then expresses the selected location as normalized text coordinates. The direct Text-2D control starts from the pretrained VLM. The learned reference-ID variant uses its corresponding object-reference warm-up, retaining learned input embeddings for reference tokens throughout training. All planning variants retain the full Navtrain set and three planning-SFT epochs.

Group-relative reinforcement fine-tuning. We initialize the policy from planning SFT and keep a frozen SFT reference for KL regularization. Each scene receives $G = 16$ sampled completions, one per GPU, with temperature 1.1, top-$p = 1$, and at most 512 new tokens. Rewards are unscaled PDM scores; advantages use a stabilizer of 10^{-4}, and scene groups with reward standard deviation below 0.05 are excluded from the update. AdamW uses learning rate 10^{-5}, no weight decay, gradient-norm clipping at 1.0, and 4,000 optimization steps. The KL coefficient is 0.04.

The policy and KL losses cover all generated completion tokens, including focus references, language plans, and waypoint text, while excluding padding and positions after EOS. For a sampled token, the KL estimator is $\exp(\Delta_{i,t}) - \Delta_{i,t} - 1$, where $\Delta_{i,t} = \log \pi_{\text{ref}}(y_{i,t} \mid x, y_{i,<t}) - \log \pi_\theta(y_{i,t} \mid x, y_{i,<t})$. Trajectory parsing pads incomplete tails when at least one waypoint is available, and reward evaluation uses the resulting trajectory. Metric-cache or scoring failures receive zero reward.

We use the on-policy log-probability loss defined in Eq. 6, without policy-ratio clipping. Advantages are fixed during each update, and both the policy and KL terms use the completion-token mask described above.

Evaluation protocol and implementation. At evaluation, we use deterministic decoding to produce one trajectory per scene, without stochastic sampling, best-of-N selection, or reranking. Our EPDMS table uses results from the updated NAVSIM implementation with human-penalty filtering enabled, including the official correction to score aggregation after filtering.[1] Some papers report earlier-version results that may not align with the updated values used in our table. We follow the updated evaluation version consistently.

[1] https://github.com/autonomousvision/navsim/issues/151

A.3.4 QUALITATIVE RESULTS

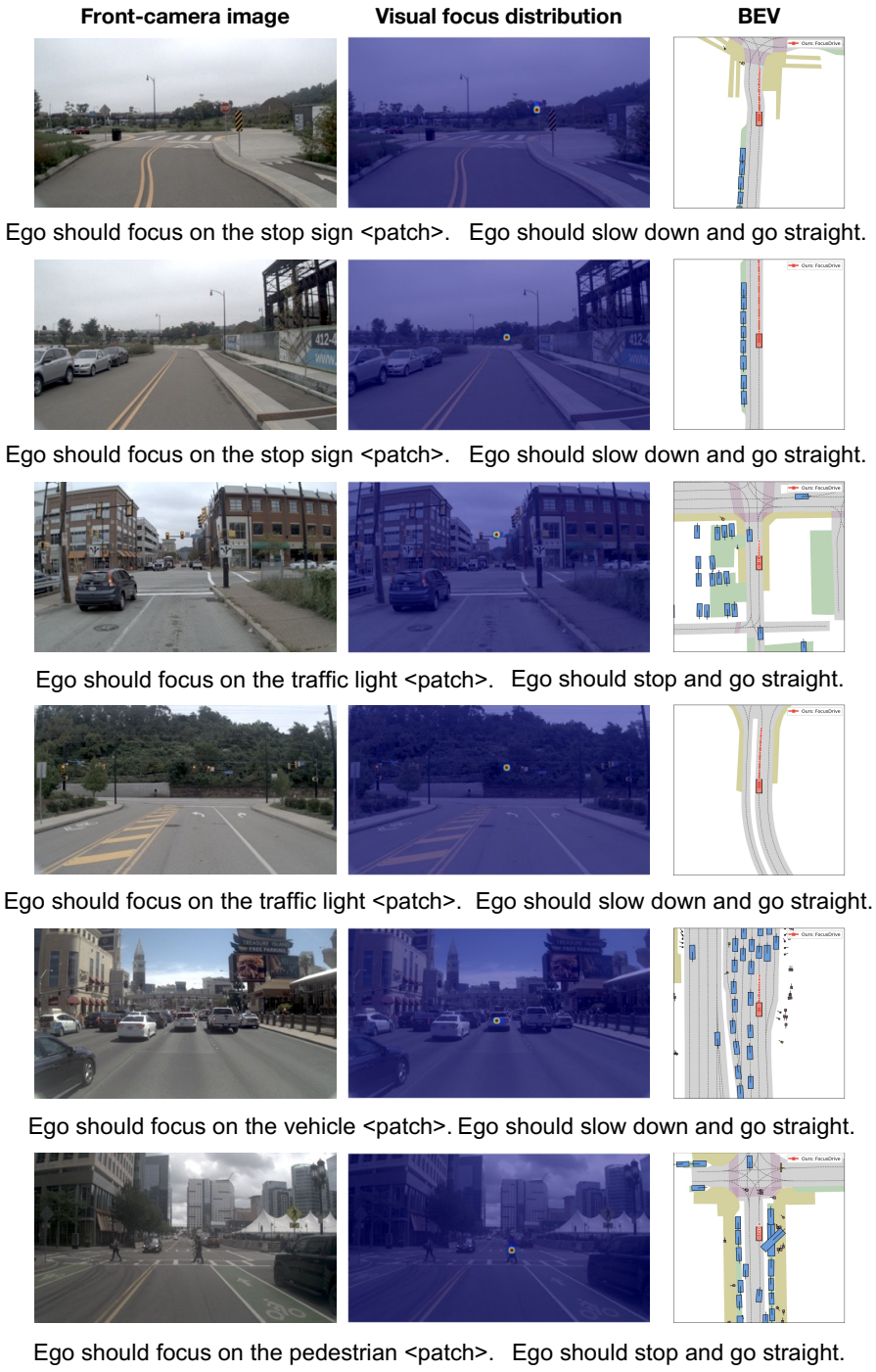

Figure 6: Additional planning examples involving stop signs, traffic lights, a vehicle, and a pedestrian. Each row pairs the front-camera image with its visual focus distribution, BEV trajectory, and generated focus and action plan. Red squares denote FocusDrive trajectories; `<patch>` denotes a selected image-patch reference. Heatmaps visualize the softmax distribution over image-patch reference logits at the reference decoding step, displayed on the patch grid with per-scene color scales.