DO BETTER VISUAL REPRESENTATIONS ALWAYS LEAD TO BETTER END-TO-END AUTONOMOUS DRIVING?

Anonymous authors
Paper under double-blind review

ABSTRACT

Visual foundation models (VFMs) are increasingly integrated into end-to-end autonomous driving for their powerful representations, yet it remains unclear when these representations improve driving performance. To investigate this question, we introduce ViRA, a planner-agnostic visual representation alignment framework that keeps the planner architecture and inference cost unchanged. Our study reveals three findings: (1) VFM-guided visual representations consistently improve driving performance across diverse end-to-end planners, with gains extending to zero-shot closed-loop evaluation. (2) The choice of VFM target matters for planning performance, and alignment to a different VFM can further benefit planners with pre-trained VFM encoders. (3) Auxiliary perception supervision reduces sensitivity to VFM target selection, narrowing the EPDMS spread across five targets from 2.7 to 0.5 points and potentially compensating for less effective VFM targets. Guided by these findings, we develop ViRA-Diffusion, a diffusion-based planner trained without auxiliary perception supervision, which achieves 92.3 EPDMS on NAVSIM v2 navtest, outperforming recent methods in our comparison by at least 1.9 points. The results motivate jointly considering target selection and planner supervision when integrating VFMs into end-to-end autonomous driving.

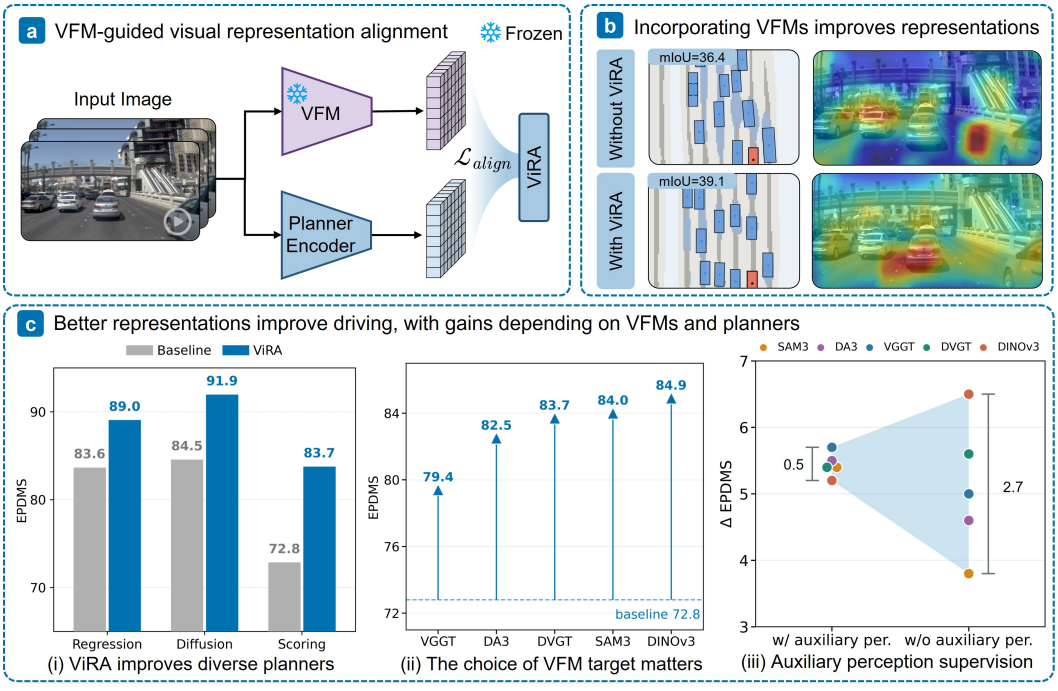

Figure 1: **Better visual representations improve planning, with gains depending on both VFMs and planners.** (a) ViRA aligns planner representations with a frozen VFM during training, leaving inference unchanged. (b) Segmentation and visual attention suggest improved representations. (c) Better visual representations improve planning, but the gains depend on the VFM target and planner.

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.

1 INTRODUCTION

End-to-end autonomous driving is considered a promising route to scalable real-world deployment (Chen et al., 2024a; Guan et al., 2026). Within this pipeline, visual representations provide the planner with information about the environment, making the learning of high-quality visual representations an important research problem (Jiang et al., 2023; Li et al., 2024). As visual foundation models (VFMs) advance in semantic understanding and 3D geometry (Siméoni et al., 2025; Wang et al., 2025; Carion et al., 2025; Lin et al., 2025), an increasing number of studies either use VFMs directly (Sima et al., 2024; Wang et al., 2026a; Naeinian et al., 2026) or incorporate their pre-trained representations through distillation and representation alignment (Strong et al., 2026; Li et al., 2026; Xie et al., 2026). However, most studies focus on a single planner–VFM pairing (Zuo et al., 2026b; Wang et al., 2026b), and systematic comparisons of planning gains across different pairings remain limited. It remains unclear whether the advantages of VFMs on visual tasks consistently translate into improved driving performance.

To answer this question, we introduce ViRA, a VFM-guided representation alignment framework illustrated in Figure 1(a). ViRA aligns the planner's intermediate features with those of a frozen VFM target while keeping the planner architecture and inference cost unchanged (Yu et al., 2025b; Singh et al., 2026; Li et al., 2026). Using this framework, we compare different combinations of end-to-end planners and VFM targets on NAVSIM v2 (Cao et al., 2025) and HUGSIM (Zhou et al., 2025a). We find that VFM-guided visual representations consistently improve driving performance across the diverse planners, but the gains depend on both the VFM target and planner design. We further examine these findings below, focusing on overall driving gains, the effects of VFM target selection, and the role of auxiliary perception supervision.

VFM-guided visual representations improve end-to-end driving performance. As shown in Figure 1(c-i), ViRA improves driving performance across regression-based (Chitta et al., 2023), diffusion-based (Liao et al., 2025), and scoring-based (Feng et al., 2026) planners, yielding average gains of 7.9 and 8.6 EPDMS points over their respective baselines on NAVSIM v2 navtest and navhard, respectively. These benefits extend to zero-shot closed-loop evaluation on HUGSIM, where overall HD-Score improves by an average of 4.0 points. Figure 1(b) shows that driving gains are accompanied by improved perception performance. To test whether the gains come from pre-training rather than loss regularization alone, we compare pre-trained and randomly initialized targets under the same alignment objectives. Randomly initialized targets fail to reproduce these perception and planning gains, supporting the importance of pre-trained VFM representations. This raises the question of whether VFMs with different pre-training objectives yield different planning benefits.

The choice of VFM target matters. As shown in Figure 1(c-ii), gains for the same Rap* (Feng et al., 2026) architecture range from 6.6 to 12.1 EPDMS points across VFM targets (a 5.5-point spread), with DINOv3 (Siméoni et al., 2025) yielding the largest gain. Yet even when Rap* uses DINOv3 as its pre-trained backbone, alignment to VGGT (Wang et al., 2025), a 3D geometric foundation model, further improves EPDMS from 83.8 to 86.3 (+2.5 points). This dependence on the VFM target also appears in TransFuser (Chitta et al., 2023) when trained without auxiliary perception supervision, with DINOv3 again performing best. With auxiliary perception supervision, however, TransFuser and DiffusionDrive (Liao et al., 2025) achieve similar planning performance across these targets. We hypothesize that auxiliary perception supervision affects how planners learn from VFM targets.

Auxiliary perception supervision reduces sensitivity to VFM target selection. To examine this factor, we compare TransFuser with and without auxiliary perception supervision. Across the same five VFM targets, planning gains over the respective baselines without representation alignment range from 5.2 to 5.7 EPDMS points with auxiliary supervision and from 3.8 to 6.5 points without it. As shown in Figure 1(c-iii), removing auxiliary supervision widens the spread in gains across targets from 0.5 to 2.7 EPDMS points. More specifically, the highest score stays nearly unchanged, while the lowest drops from 88.8 to 86.4, indicating that auxiliary supervision mainly compensates for the less effective targets. These results show that, in this setting, auxiliary perception supervision narrows the differences in gains across alignment targets, making planning performance less sensitive to VFM target selection.

These findings make a concrete prediction: a planner can achieve strong planning performance when its auxiliary perception supervision is removed, and a suitable VFM target is supplied in its place. We verify this with **ViRA-Diffusion**, a diffusion-based (Liao et al., 2025) planner trained with ViRA

using DINOv3 (Siméoni et al., 2025) as the target and no auxiliary perception supervision. It reaches 92.3 EPDMS on NAVSIM v2 (Cao et al., 2025) navtest, outperforming the recent state-of-the-art method by 1.9 points. We highlight the main contributions of this paper below:

- We systematically study the driving benefits of pre-trained VFM representations across planners and targets using ViRA, a planner-agnostic experimental framework that preserves the deployed planner architecture and inference cost.
- We show that VFM target selection affects planning performance, while auxiliary perception supervision narrows performance differences across targets, highlighting the importance of considering target selection and planner supervision jointly.
- We demonstrate the practical applicability of VFM-guided alignment through **ViRA-Diffusion**, which uses DINOv3 alignment to achieve 92.3 EPDMS on NAVSIM v2 navtest without auxiliary perception supervision.

2 Preliminaries

2.1 VFM-Guided End-to-End Planner

End-to-end Planner. Given a single-frame multi-view observation $\mathcal{O} = \{\mathbf{I}^i\}_{i=1}^N$, an end-to-end planner predicts a future ego trajectory $\hat{\tau} = \{\hat{\mathbf{p}}_k\}_{k=1}^K$, where N is the number of camera views and $\hat{\mathbf{p}}_k = (\hat{x}_k, \hat{y}_k)$ denotes the predicted position at future step k in the ego coordinate system. We formulate the planner as an encoder E_θ followed by a policy head π_ψ:

$$\mathbf{F}^s = E_\theta(\mathcal{O}), \qquad \hat{\tau} = \pi_\psi(\mathbf{F}^s). \tag{1}$$

The encoder maps visual observations into an intermediate representation $\mathbf{F}^s$, from which the policy head extracts information for trajectory planning. The policy π_ψ can follow regression-based (Chitta et al., 2023), diffusion-based (Liao et al., 2025), or scoring-based (Feng et al., 2026) designs. This formulation provides a common interface for studying visual representations across planners with different trajectory prediction mechanisms.

VFM-Guided Training. To incorporate powerful VFM representations into planner representation learning, we design a VFM-guided training pipeline. During training, the same observation $\mathcal{O}$ is fed to a frozen VFM T_ϕ to obtain $\mathbf{F}^t = T_\phi(\mathcal{O})$. These representations provide representation-level supervision for the planner encoder without being passed to the policy head π_ψ or requiring additional annotations for planner training. Only the original planner is kept at inference, leaving its deployed architecture and inference cost unchanged. Unless otherwise specified, we use DVGT (Zuo et al., 2026a) as T_ϕ, while the same pipeline also supports other VFMs (Carion et al., 2025; Wang et al., 2025; Siméoni et al., 2025; Lin et al., 2025). Throughout the paper, DINOv3 refers to DINOv3-L unless otherwise specified.

Visual Representation Alignment. To support the spatial reasoning and scene understanding required for end-to-end autonomous driving, we design two objectives of ViRA to transfer high-quality VFM representations to the planner encoder: Spatial Dependency Alignment (SDA) and Scene Semantic Alignment (SSA). SDA matches spatial attention distributions between $\mathbf{F}^s$ and $\mathbf{F}^t$, encouraging the planner to learn the spatial dependencies encoded by the VFM. SSA aligns their global embeddings using a bidirectional contrastive objective, encouraging scene-level semantic consistency between the planner and the VFM. The overall training objective is

$$\mathcal{L} = \mathcal{L}_{\text{planner}} + \mathcal{L}_{\text{align}}, \quad \mathcal{L}_{\text{align}} = \lambda_{\text{SDA}}\mathcal{L}_{\text{SDA}} + \lambda_{\text{SSA}}\mathcal{L}_{\text{SSA}}. \tag{2}$$

where $\mathcal{L}_{\text{planner}}$ is the original planner objective, including trajectory prediction and any auxiliary supervision used in the corresponding configuration, $\mathcal{L}_{\text{align}}$ is the representation alignment objective combining SDA and SSA, and λ_{SDA} and λ_{SSA} are scalar weights. Details of ViRA are provided in Appendix A.

2.2 Benchmark and Evaluation

NAVSIM (Dauner et al., 2024) is a data-driven, non-reactive simulation and benchmarking framework for autonomous driving built on OpenScene (OpenScene Contributors, 2023) and nuPlan (Karnchanachari et al., 2024). We use navtrain for training and evaluate models on two NAVSIM v2 (Cao

et al., 2025) benchmarks, navtest and navhard. Both benchmarks adopt the rule-based planning metric EPDMS (Li et al., 2025a), whose sub-metrics include No-at-fault Collisions (NC), Drivable Area Compliance (DAC), Driving Direction Compliance (DDC), Traffic Light Compliance (TLC), Time-to-Collision (TTC), Ego Progress (EP), Lane Keeping (LK), History Comfort (HC), and Extended Comfort (EC).

HUGSIM (Zhou et al., 2025a) is a real-time, photorealistic closed-loop simulator that reconstructs 3D scenes from captured RGB images with 3D Gaussian Splatting and updates ego and actor states at each step from the planner output, so that surrounding vehicles react to the ego. We evaluate planners trained only on NAVSIM in a zero-shot manner on scenarios grouped into four difficulty levels: Easy, Medium, Hard, and Extreme. The evaluation metric is HD-Score, which weights route completion (RC) by per-step safety and comfort scores. Beyond planning scores, we use mean Intersection-over-Union (mIoU) over seven BEV semantic classes to measure TransFuser's perception performance and assess the quality of its learned representations. More details are provided in Appendix B.

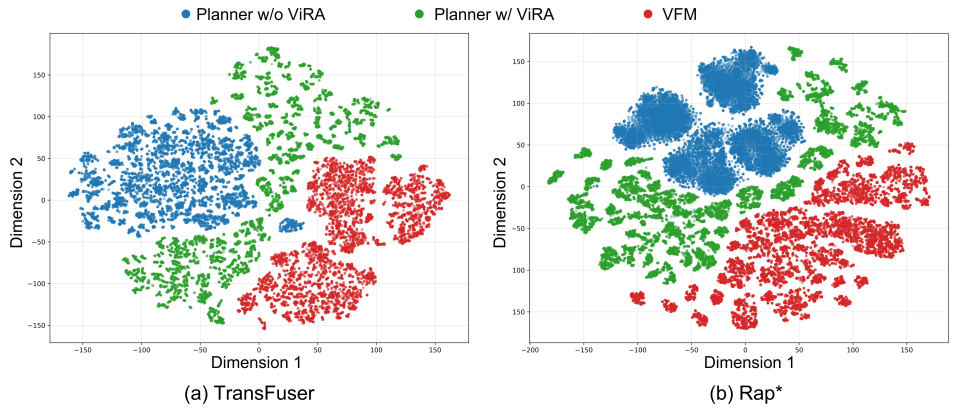

Figure 2: **t-SNE visualization of visual representations.** Planner representations without and with ViRA are shown alongside the reference VFM representations for (a) TransFuser and (b) Rap*. The visualization suggests that representation alignment with ViRA shifts planner representations toward those of the reference VFM in the t-SNE space.

Table 1: **Performance on the NAVSIM v2 navtest benchmark.** Rap* is our reimplementation with a different backbone from the official codebase and without the original data augmentation. For a fair comparison, all methods use camera input only, without LiDAR.

Methods	Decoder	NC ↑	DAC ↑	DDC ↑	TLC ↑	EP ↑	TTC ↑	LK ↑	HC ↑	EC ↑	EPDMS ↑
Human Agent	-	100	100	99.8	100	97.8	100	100	98.1	90.1	97.9
Perception-based											
TransFuser (Chitta et al., 2023)	Regression	97.1	89.6	99.0	99.8	97.9	95.9	95.9	98.3	87.3	83.6
TransFuser-ViRA		97.7	94.1	99.5	99.8	98.2	96.7	96.8	98.3	87.9	**89.0** +5.4
DiffusionDrive (Liao et al., 2025)	Diffusion	98.2	95.9	99.4	99.8	87.5	97.3	96.8	98.3	87.7	84.5
DiffusionDrive-ViRA		98.2	96.5	99.6	99.8	98.2	97.4	97.4	98.3	88.0	**91.9** +7.4
Perception-free											
Rap* (Feng et al., 2026)	Scoring	91.6	88.8	93.5	99.8	87.2	92.5	90.9	97.2	56.7	72.8
Rap*-ViRA		97.1	96.9	99.2	99.9	87.5	96.9	96.2	97.7	48.1	**83.7** +10.9

3 VFM-GUIDED VISUAL REPRESENTATIONS DO IMPROVE DRIVING

3.1 DRIVING PERFORMANCE ACROSS DIFFERENT BENCHMARKS

Results on NAVSIM v2 navtest. Table 1 shows that VFM-guided representation alignment consistently improves driving performance across three representative planners. With ViRA, TransFuser (Chitta et al., 2023), DiffusionDrive (Liao et al., 2025), and Rap* (Feng et al., 2026) achieve

Table 2: **Performance on the NAVSIM v2 navhard benchmark.** Rap* is our reimplementation with a different backbone from the official codebase and without the original data augmentation. PDM-Closed plans with GT symbolic inputs, while all other methods use camera data only as input.

Methods	Decoder	Stage	NC ↑	DAC ↑	DDC ↑	TLC ↑	EP ↑	TTC ↑	LK ↑	HC ↑	EC ↑	EPDMS ↑
PDM-Closed	-	S1	94.4	98.8	100	99.5	100	93.5	99.3	87.7	36.0	56.6
		S2	90.5	90.6	95.4	98.4	100	98.4	74.2	91.9	29.7	
Perception-based												
TransFuser (Chitta et al., 2023)	Regression	S1	96.4	73.8	95.6	99.3	94.8	95.6	88.0	97.6	82.2	27.6
		S2	82.6	62.8	77.9	98.2	91.0	79.9	40.9	97.5	80.8	
TransFuser-ViRA		S1	97.4	80.0	98.7	99.3	98.0	95.6	95.8	97.8	81.3	**31.7** +4.1
		S2	81.2	73.9	86.0	98.0	97.0	78.3	47.3	96.1	71.1	
DiffusionDrive (Liao et al., 2025)	Diffusion	S1	96.8	86.0	98.8	99.3	84.0	95.8	96.7	97.6	79.6	30.5
		S2	80.1	72.8	84.4	98.4	85.9	76.6	46.4	96.3	72.8	
DiffusionDrive-ViRA		S1	96.4	87.8	99.2	99.3	98.5	95.3	96.4	97.6	78.7	**35.8** +5.3
		S2	80.5	74.1	86.4	98.4	97.5	77.2	47.6	95.8	71.3	
Perception-free												
Rap* (Feng et al., 2026)	Scoring	S1	89.4	71.8	82.3	99.6	86.6	89.6	85.8	95.8	47.1	32.8
		S2	90.6	77.6	85.6	98.8	71.4	88.9	50.8	98.4	68.9	
Rap*-ViRA		S1	96.7	90.7	97.2	100.0	85.2	96.2	94.2	96.4	42.2	**49.2** +16.4
		S2	95.5	87.6	94.9	99.6	69.3	94.5	55.3	98.7	68.3	

Table 3: **Zero-shot Performance on the HUGSIM Benchmark.** Rap* is our reimplementation with a different backbone from the official codebase and without the original data augmentation.

Method	Easy		Medium		Hard		Extreme		Overall	
	RC	HD-Score	RC	HD-Score	RC	HD-Score	RC	HD-Score	RC	HD-Score
Perception-based										
TransFuser (Chitta et al., 2023)	66.8	56.7	28.1	11.6	24.5	9.2	26.2	10.9	36.4	22.1
TransFuser-ViRA	72.7	62.6	31.9	15.3	27.8	11.4	25.5	9.9	39.5	**24.8** +2.7
DiffusionDrive (Liao et al., 2025)	68.9	56.8	29.8	12.4	25.9	9.7	25.7	9.8	37.6	22.2
DiffusionDrive-ViRA	77.5	69.3	37.7	20.7	30.0	12.0	27.9	12.3	43.3	**28.6** +6.4
Perception-free										
Rap* (Feng et al., 2026)	11.8	10.6	11.1	9.9	7.8	6.3	4.1	2.3	8.7	7.3
Rap*-ViRA	18.1	15.9	16.7	14.6	10.5	8.2	4.3	2.3	12.4	**10.3** +3.0

EPDMS gains of 5.4, 7.4, and 10.9 points over their respective baselines. DAC and LK show substantial improvements, reflecting gains in drivable-area compliance and lane keeping, respectively. Both capabilities are closely tied to visual perception. These improvements suggest that the benefits of VFM-guided visual representations extend across different planner architectures in standard driving scenarios.

Results on NAVSIM v2 navhard. The benefits extend to challenging driving scenarios: all three planners achieve higher EPDMS with ViRA on navhard (Table 2). Notably, although Rap* already achieves the highest baseline EPDMS among the three planners, representation alignment further improves its score from 32.8 to 49.2, a gain of 16.4 points. DAC and DDC also improve, reflecting better drivable-area compliance and driving direction compliance in challenging scenarios. These improvements highlight the value of VFM-guided visual representations even for the strongest evaluated planner in challenging scenarios.

Zero-shot Results on HUGSIM. To assess whether the gains extend beyond NAVSIM, we evaluate NAVSIM-trained planners on the closed-loop HUGSIM benchmark (Zhou et al., 2025a) without further training. As shown in Table 3, ViRA improves the overall HD-Score of TransFuser, DiffusionDrive and Rap* by 2.7, 6.4 and 3.0 points, respectively. Together with the NAVSIM results, these zero-shot gains support our finding that VFM-guided visual representations do improve driving performance across different benchmarks and evaluation settings.

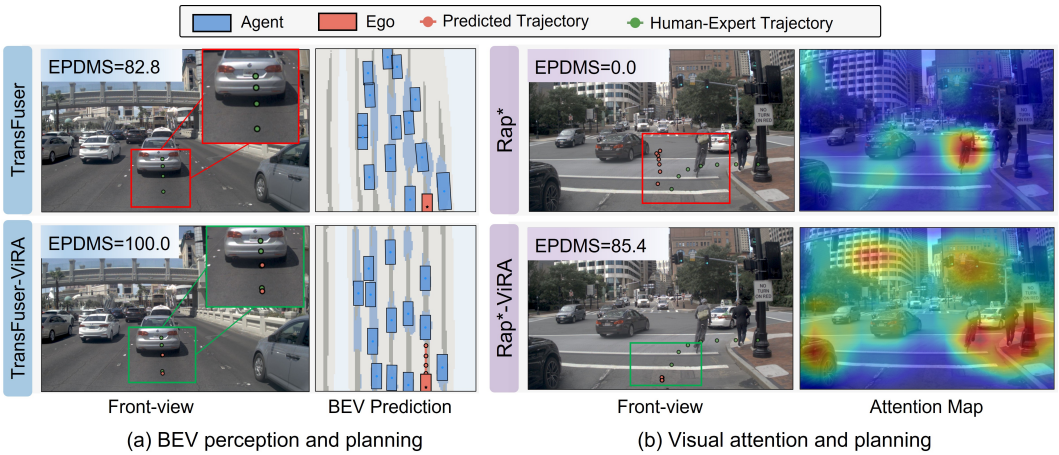

Figure 3: **Qualitative results showing improved perception and planning with ViRA.**

3.2 Representation Quality Improves Measurably

To examine whether the driving gains are accompanied by improved perception, we analyze BEV segmentation and attention maps. Figure 5(a) shows that ViRA improves TransFuser's BEV segmentation mIoU by 0.5–3.0 points across the evaluated VFM targets. The t-SNE visualizations in Figure 2 suggest that ViRA shifts the representations of TransFuser and Rap* toward those of the DVGT target in the t-SNE embedding space. The qualitative examples in Figure 3 further illustrate improvements in BEV predictions and planned trajectories, together with changes in attention to surrounding road users. These results provide evidence that alignment with high-quality VFM representations improves perception alongside driving performance.

Table 4: **VFMs dependency on BEV segmentation of TransFuser.** Static objects and pedestrians are omitted from the class-wise results; mIoU is computed over all seven classes.

VFM	Background ↑	Road ↑	Walkway ↑	Centerline ↑	Vehicle ↑	mIoU ↑
Baseline	85.4	62.3	56.3	23.6	27.1	36.4
Random-init	82.4	58.3	47.2	12.5	14.7	30.7 -5.7
DVGT (Zuo et al., 2026a)	86.9	64.6	59.3	28.9	33.4	**39.1** +2.7

Table 5: **Effect of different VFM teachers on TransFuser and Rap*.** Performance on the NAVSIM v2 navtest benchmark.

Methods	VFMs	NC ↑	DAC ↑	DDC ↑	TLC ↑	LK ↑	EPDMS ↑
Perception-based							
TransFuser (Chitta et al., 2023)	Baseline	97.1	89.6	99.0	99.8	95.9	83.6
	Random-init	96.8	90.2	99.0	99.7	95.5	82.0 -1.6
	DVGT (Zuo et al., 2026a)	97.7	94.1	99.5	99.8	96.8	**89.0** +5.4
Perception-free							
Rap* (Feng et al., 2026)	Baseline	91.6	88.8	93.5	99.8	90.9	72.8
	Random-init	94.6	88.1	95.1	99.8	93.1	71.1 -1.7
	DVGT (Zuo et al., 2026a)	97.1	96.9	99.2	99.9	96.2	**83.7** +10.9

3.3 The Gains Require a Pre-trained Target

To test whether the gains depend on VFM pre-training, we replace the pre-trained target with a randomly initialized one while retaining the same alignment objectives. As shown in Table 4, random-target alignment reduces TransFuser's BEV segmentation mIoU from 36.4 to 30.7, whereas alignment with pre-trained DVGT increases it to 39.1. The same pattern holds for driving performance: random-target alignment lowers EPDMS by 1.6 points for TransFuser and 1.7 points for Rap*, while the

pre-trained target improves both planners (Table 5). These results support the importance of pre-trained visual knowledge: applying the same alignment objectives to random representations does not reproduce the gains observed with pre-trained VFM representations.

3.4 Gains without Additional Inference Cost

A practical question is whether these representation gains require replacing the planner encoder with a larger VFM at inference. Figure 4 compares ViRA with direct VFM backbone replacement in terms of planning performance, inference latency, and parameter count. Latency is measured on a single NVIDIA H100 GPU. DVGT-guided alignment improves EPDMS from 72.8 to 83.7 while retaining the baseline latency of 20 ms and parameter count of 69M. This nearly matches the strongest evaluated backbone replacement, DINOv3-L, which achieves 83.8 EPDMS but requires 65 ms and 351M parameters. Since the VFM and alignment heads are used only during training, ViRA enables the planner to benefit from pre-trained visual representations without changing its deployed architecture or increasing inference cost.

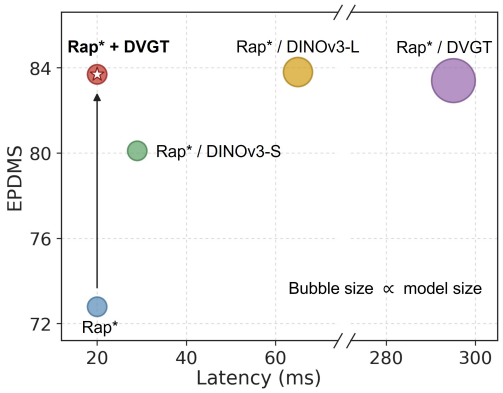

Figure 4: **ViRA (star) vs. backbone replacement (circles).** Bubble size indicates parameter count.

4 Planning Gains Depend on Both VFMs and Planners

4.1 The choice of VFM target matters

Table 6: **Rap* (Feng et al., 2026) with DINOv3 (Siméoni et al., 2025) backbones.** Performance on the NAVSIM v2 navtest benchmark under different VFM teachers.

Backbone	Teachers	NC ↑	DAC ↑	DDC ↑	TLC ↑	LK ↑	EPDMS ↑
DINOv3-vit16l	-	97.2	96.4	98.9	99.9	96.1	83.8
DINOv3-vit16l	VGGT (Wang et al., 2025)	98.5	97.3	99.3	100.0	96.7	**86.3** +2.5

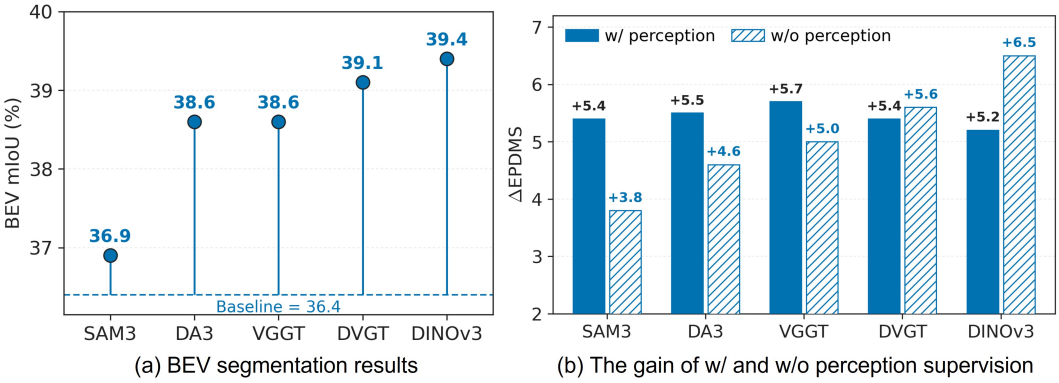

Figure 5: **Auxiliary perception supervision narrows the spread of planning gains across VFM targets.** (a) BEV mIoU across VFM targets; the dashed line marks the baseline (36.4). (b) Planning gain ΔEPDMS with and without auxiliary perception supervision (baselines 83.6 and 82.6, respectively).

As shown in Figure 1(c-ii), changing only the representation alignment target for the perception-free Rap* yields EPDMS scores ranging from 79.4 to 84.9, a spread of 5.5 points. This difference

is comparable to the gains from many method-level improvements. These results highlight the importance of alignment target selection, with DINOv3, which learns from more general data, providing the largest planning gain in this setting. To further examine whether VFMs with different pre-training objectives provide complementary representations, we use Rap* with a DINOv3-L backbone as the baseline and introduce alignment to VGGT, a VFM trained for 3D geometry.

As shown in Table 6, VGGT alignment improves EPDMS by 2.5 points, from 83.8 to 86.3. Thus, even when the backbone already provides strong semantic representations, alignment to a geometry-focused VFM can further improve planning performance. These results suggest that alignment target selection deserves careful consideration, as its benefits may depend on the complementarity between the target VFM and the planner's existing representations.

4.2 Auxiliary Supervision Shapes Sensitivity to VFM Targets

Extending the experiments from the preceding subsection to perception-based planners, we find that TransFuser and DiffusionDrive are less sensitive to the choice of VFM target. Across VFM targets, EPDMS varies by only 0.5 points for TransFuser (88.8 to 89.3) and 0.6 points for DiffusionDrive (91.3 to 91.9), compared with 5.5 points for the perception-free Rap*. One important difference between these planners is the use of auxiliary perception supervision. TransFuser and DiffusionDrive learn planner representations with BEV segmentation and agent detection supervision, whereas Rap* does not. We therefore hypothesize that auxiliary perception supervision reduces the sensitivity of the planner to VFM target selection.

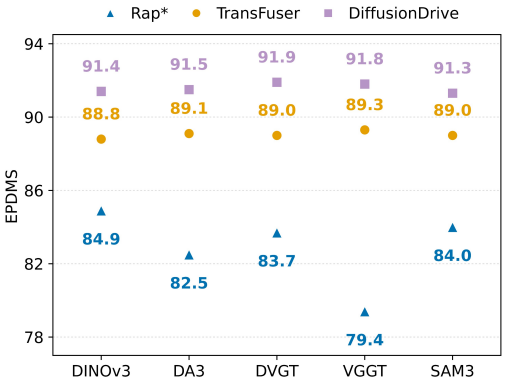

Figure 6: **Different VFM targets across planners.**

To test this hypothesis, we compare TransFuser (Chitta et al., 2023) with and without auxiliary perception supervision across five alignment targets: DVGT (Zuo et al., 2026a), VGGT (Wang et al., 2025), DA3 (Lin et al., 2025), DINOv3 (Siméoni et al., 2025), and SAM3 (Carion et al., 2025). As shown in Figure 5(b), EPDMS ranges from 88.8 to 89.3 with auxiliary supervision. Removing this supervision expands the range to 86.4 to 89.1, increasing the spread from 0.5 to 2.7 points. Examining the endpoints of these ranges provides further insight: after removing auxiliary supervision, the maximum score remains nearly unchanged (89.3 to 89.1), whereas the minimum drops substantially (88.8 to 86.4). This indicates that auxiliary supervision mainly compensates for the less effective targets. Across these targets, TransFuser's BEV segmentation results under auxiliary supervision show a broadly similar ordering to its planning results without auxiliary supervision. Overall, these controlled comparisons show that auxiliary perception supervision reduces TransFuser's sensitivity to VFM target selection.

Table 7: **Comparison with state-of-the-art methods on the NAVSIM v2 navtest benchmark.** ViRA-Diffusion is DiffusionDrive (Liao et al., 2025) trained with ViRA and without auxiliary perception supervision.

Method	**NC ↑**	**DAC ↑**	**DDC ↑**	**TLC ↑**	**EP ↑**	**TTC ↑**	**LK ↑**	**HC ↑**	**EC ↑**	**EPDMS ↑**
Epona (Zhang et al., 2025)	97.1	95.7	99.3	99.7	88.6	96.3	97.0	98.0	67.8	85.1
DiffusionDriveV2 (Zou et al., 2025)	97.7	96.6	99.2	99.8	88.9	97.2	96.0	97.8	91.0	87.5
Latent-WAM (Wang et al., 2026b)	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)	98.8	99.1	99.6	99.9	86.6	98.4	96.4	98.3	74.8	89.9
SparseDriveV2 (Sun et al., 2026)	98.1	98.1	99.6	99.8	91.1	97.3	96.9	98.2	78.4	90.1
Discrete-WAM (Yao et al., 2026)	98.5	98.2	99.7	99.8	90.5	97.9	97.2	98.3	78.1	90.4
ViRA-Diffusion	98.3	96.8	99.5	99.8	98.2	97.5	97.4	98.3	88.0	**92.3**

4.3 Practical Implications

The preceding analysis suggests that a planner can maintain comparable planning performance without auxiliary perception supervision when aligned with a suitable VFM target. Building on these findings, we develop ViRA-Diffusion, a diffusion-based planner (Liao et al., 2025) trained with ViRA using DINOv3 as the target and no auxiliary perception supervision. As shown in Table 7, the resulting ViRA-Diffusion achieves 92.3 EPDMS on NAVSIM v2 navtest, comparing favorably with the recent results included in our comparison. This result illustrates the practical value of our analysis in guiding the joint choice of VFM targets and supervision strategies for end-to-end planners.

5 Related Work

End-to-End Autonomous Driving. End-to-end autonomous driving learns planning and control directly from sensory inputs within a unified framework (Hu et al., 2023; Yu et al., 2025a; Sun et al., 2025). Joint optimization can mitigate cascading errors between separately trained modules and facilitate learning from large-scale driving data, making end-to-end autonomous driving a promising route toward scalable real-world deployment (Chen et al., 2024a; Guan et al., 2026). Recent studies focus on improving policy heads (Liao et al., 2025; Gao et al., 2025; Zheng et al., 2026), introducing auxiliary perception supervision (Strong et al., 2026; Chen et al., 2024b; Xing et al., 2025), and scaling training data through synthetic scene generation and augmentation (Zhou et al., 2025b; Xu et al., 2025; Li et al., 2025b; Tian et al., 2026). Despite this progress, how visual representation quality contributes to planning gains remains insufficiently understood. Some recent methods enhance visual representations for driving with semantic and geometric priors from pre-trained VFMs (Zheng et al., 2025; Wang et al., 2026b). These methods demonstrate benefits within specific systems, but provide limited guidance on how the gains vary across VFM targets and planner designs. Complementing these directions, we study when better visual representations improve planning performance.

Representation Learning for Driving. Learning visual representations that capture scene semantics and spatial structure is crucial for end-to-end autonomous driving (Chen et al., 2024a). To enrich these representations, recent driving studies turn to VFMs pre-trained on large-scale visual data, drawing on their capabilities in semantics (Kirillov et al., 2023; Carion et al., 2025), depth (Yang et al., 2024a;b; Lin et al., 2025), and 3D geometry (Zuo et al., 2026a; Wang et al., 2024; 2025). These VFMs are used through visual backbone replacement (Feng et al., 2026; Naeinian et al., 2026), auxiliary supervision from VFM predictions (Jia et al., 2025; Strong et al., 2026), or feature integration within task-specific architectures and training objectives (Zheng et al., 2025; Wang et al., 2026b;a; Xie et al., 2026). Recent studies show that representation alignment, which supervises intermediate features with frozen visual representations, can improve training efficiency and task performance (Yu et al., 2025b; Li et al., 2026). iREPA (Singh et al., 2026) further shows that the spatial structure of target representations predicts image generation quality better than their ImageNet classification accuracy. In end-to-end autonomous driving, how planner architecture and auxiliary perception supervision shape the benefits of VFM-guided representations remains unclear. We use representation alignment as a controlled experimental interface to study these interactions across planners and VFM targets.

6 Conclusion and Outlook

In this paper, we systematically study when VFM-guided visual representations improve end-to-end autonomous driving, using ViRA as an experimental interface that keeps the planner's architecture and inference cost unchanged. Experiments across planners and benchmarks show consistent driving gains, with benefits extending to zero-shot closed-loop evaluation. VFM target selection affects these gains, and alignment to a different VFM can further improve planners that already use pre-trained VFM backbones. Controlled comparisons on TransFuser further show that auxiliary perception supervision narrows performance differences across VFM targets. These results highlight the importance of considering VFM target selection and planner supervision jointly. Guided by this analysis, we develop ViRA-Diffusion with DINOv3 alignment and no auxiliary perception supervision, achieving 92.3 EPDMS on NAVSIM v2 navtest. Future work will examine whether these findings extend to additional planner architectures and other physical intelligence tasks involving decision-making.

AI USE STATEMENT

In this work, we did use a generative AI tool (ChatGPT) for two subtasks with recommended disclosure: (i) literature search and summarization, to help locate and summarize related work; and (ii) language editing, to improve the clarity and readability of author-written text. We have reviewed all AI-assisted work: every reference surfaced by the tool was retrieved and read in its original form before being cited, and all AI-edited text was checked by the authors against the intended meaning of the original draft. We take responsibility for the final content of this work, including text, claims, or artifacts produced with the aid of generative AI.

REPRODUCIBILITY STATEMENT

We provide full implementation and training details in Appendix B, including per-planner hyperparameters (Table S.1) and alignment-module details (Appendix A). All datasets and benchmarks are public (NAVSIM v2 (Cao et al., 2025), HUGSIM (Zhou et al., 2025a)), all VFM targets use official public checkpoints (Siméoni et al., 2025; Wang et al., 2025; Zuo et al., 2026a; Carion et al., 2025; Lin et al., 2025), and all planners use public resources (Chitta et al., 2023; Liao et al., 2025; Feng et al., 2026). We plan to release code, training scripts, and model checkpoints upon publication to facilitate reproduction.

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. Pseudosimulation for autonomous driving. In *CoRL*, 2025.

Nicolas Carion, Laura Gustafson, Yuan-Ting Hu, Shoubhik Debnath, Ronghang Hu, Didac Suris, Chaitanya Ryali, Kalyan Vasudev Alwala, Haitham Khedr, Andrew Huang, et al. Sam 3: Segment anything with concepts. *arXiv preprint arXiv:2511.16719*, 2025.

Li Chen, Penghao Wu, Kashyap Chitta, Bernhard Jaeger, Andreas Geiger, and Hongyang Li. End-to-end autonomous driving: Challenges and frontiers. *IEEE TPAMI*, 2024a.

Shaoyu Chen, Bo Jiang, Hao Gao, Bencheng Liao, Qing Xu, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. Vadv2: End-to-end vectorized autonomous driving via probabilistic planning. *arXiv preprint arXiv:2402.13243*, 2024b.

Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *ICML*, 2020.

Kashyap Chitta, Aditya Prakash, Bernhard Jaeger, Zehao Yu, Katrin Renz, and Andreas Geiger. Transfuser: Imitation with transformer-based sensor fusion for autonomous driving. *IEEE TPAMI*, 2023.

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 nonreactive autonomous vehicle simulation and benchmarking. In *NeurIPS*, 2024.

Lan Feng, Yang Gao, Eloi Zablocki, Quanyi Li, Wuyang Li, Sichao Liu, Matthieu Cord, and Alexandre Alahi. Rap: 3d rasterization augmented end-to-end planning. In *ICLR*, 2026.

Hao Gao, Shaoyu Chen, Bo Jiang, Bencheng Liao, Yiang Shi, Xiaoyang Guo, Yuechuan Pu, Haoran Yin, Xiangyu Li, Xinbang Zhang, Ying Zhang, Wenyu Liu, Qian Zhang, and Xinggang Wang. Rad: Training an end-to-end driving policy via large-scale 3dgs-based reinforcement learning. In *NeurIPS*, 2025.

Yanchen Guan, Xingcheng Liu, Bin Rao, Chengyue Wang, Guofa Li, Yunjian Li, Lishengsa Yue, Zhiyong Cui, Chengzhong Xu, and Zhenning Li. Planning-oriented end-to-end autonomous driving: Architectures, evaluation, and emerging paradigms, 2026.

Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *CVPR*, 2016.

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.

Yihan Hu, Jiazhi Yang, Li Chen, Keyu Li, Chonghao Sima, Xizhou Zhu, Siqi Chai, Senyao Du, Tianwei Lin, Wenhai Wang, et al. Planning-oriented autonomous driving. In *CVPR*, 2023.

Xiaosong Jia, Yanhao Liu, Junqi You, Renqiu Xia, Yu Hong, and Junchi Yan. Drivevggt: Visual geometry transformer for autonomous driving. *arXiv preprint arXiv:2511.22264*, 2025.

Bo Jiang, Shaoyu Chen, Qing Xu, Bencheng Liao, Jiajie Chen, Helong Zhou, Qian Zhang, Wenyu Liu, Chang Huang, and Xinggang Wang. Vad: Vectorized scene representation for efficient autonomous driving. In *2023 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 8306–8316. IEEE, 2023.

Napat Karnchanachari, Dimitris Geromichalos, Kok Seang Tan, Nanxiang Li, Christopher Eriksen, Shakiba Yaghoubi, Noushin Mehdipour, Gianmarco Bernasconi, Whye Kit Fong, Yiluan Guo, et al. Towards learning-based planning: The nuplan benchmark for real-world autonomous driving. In *ICRA*, 2024.

Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *ICLR*, 2015.

Alexander Kirillov, Eric Mintun, Nikhila Ravi, Hanzi Mao, Chloe Rolland, Laura Gustafson, Tete Xiao, Spencer Whitehead, Alexander C Berg, Wan-Yen Lo, et al. Segment anything. In *ICCV*, 2023.

Fuhao Li, Wenxuan Song, Han Zhao, Jingbo Wang, Pengxiang Ding, Donglin Wang, Long Zeng, and Haoang Li. Spatial forcing: Implicit spatial representation alignment for vision-language-action model. In *ICLR*, 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*, 2025a.

Tianyu Li, Yihang Qiu, Zhenhua Wu, Carl Lindström, Peng Su, Matthias Nießner, and Hongyang Li. Mtgs: Multi-traversal gaussian splatting. *arXiv preprint arXiv:2503.12552*, 2025b.

Zhiqi Li, Wenhai Wang, Hongyang Li, Enze Xie, Chonghao Sima, Tong Lu, Qiao Yu, and Jifeng Dai. Bevformer: learning bird's-eye-view representation from lidar-camera via spatiotemporal transformers. *IEEE TPAMI*, 2024.

Bencheng Liao, Shaoyu Chen, Haoran Yin, Bo Jiang, Cheng Wang, Sixu Yan, Xinbang Zhang, et al. Diffusiondrive: Truncated diffusion model for end-to-end autonomous driving. In *CVPR*, 2025.

Haotong Lin, Sili Chen, Junhao Liew, Donny Y Chen, Zhenyu Li, Guang Shi, Jiashi Feng, and Bingyi Kang. Depth anything 3: Recovering the visual space from any views. *arXiv preprint arXiv:2511.10647*, 2025.

Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *ICLR*, 2019.

Fatemeh Naeinian, Ali Hamza, Haoran Zhu, and Anna Choromanska. Zero-shot cross-city generalization in end-to-end autonomous driving: Self-supervised versus supervised representations. *arXiv preprint arXiv:2603.11417*, 2026.

Aaron van den Oord, Yazhe Li, and Oriol Vinyals. Representation learning with contrastive predictive coding. *arXiv preprint arXiv:1807.03748*, 2018.

OpenScene Contributors. Openscene: The largest up-to-date 3d occupancy prediction benchmark in autonomous driving. https://github.com/OpenDriveLab/OpenScene, 2023.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021.

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 *ECCV*, 2024.

Oriane Siméoni, Huy V Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, et al. Dinov3. *arXiv preprint arXiv:2508.10104*, 2025.

Jaskirat Singh, Xingjian Leng, Zongze Wu, Liang Zheng, Richard Zhang, Eli Shechtman, and Saining Xie. What matters for representation alignment: Global information or spatial structure? In *ICLR*, 2026.

Matthew Strong, Wei-Jer Chang, Quentin Herau, Jiezhi Yang, Yihan Hu, Chensheng Peng, and Wei Zhan. Learning to drive is a free gift: Large-scale label-free autonomy pretraining from unposed in-the-wild videos. In *CVPR*, 2026.

Wenchao Sun, Xuewu Lin, Yining Shi, Chuang Zhang, Haoran Wu, and Sifa Zheng. Sparsedrive: End-to-end autonomous driving via sparse scene representation. In *ICRA*, 2025.

Wenchao Sun, Xuewu Lin, Keyu Chen, Zixiang Pei, Xiang Li, Yining Shi, and Sifa Zheng. Sparsedrive-v2: Scoring is all you need for end-to-end autonomous driving, 2026.

Haochen Tian, Tianyu Li, Haochen Liu, Jiazhi Yang, Yihang Qiu, Guang Li, Junli Wang, Yinfeng Gao, Zhang Zhang, Liang Wang, Hangjun Ye, Long Chen, and Hongyang Li. Simscale: Learning to drive via real-world simulation at scale. In *CVPR*, 2026.

Frederick Tung and Greg Mori. Similarity-preserving knowledge distillation. In *ICCV*, 2019.

Jianyuan Wang, Minghao Chen, Nikita Karaev, Andrea Vedaldi, Christian Rupprecht, and David Novotny. Vggt: Visual geometry grounded transformer. In *CVPR*, 2025.

Jie Wang, Guang Li, Zhijian Huang, Chenxu Dang, Hangjun Ye, Yahong Han, and Long Chen. Vgdrive: Empowering vision-language models with cross-view geometric grounding for autonomous driving. In *CVPR*, 2026a.

Linbo Wang, Yupeng Zheng, Qiang Chen, Shiwei Li, Yichen Zhang, Zebin Xing, Qichao Zhang, Xiang Li, Deheng Qian, Pengxuan Yang, et al. Latent-wam: Latent world action modeling for end-to-end autonomous driving. *arXiv preprint arXiv:2603.24581*, 2026b.

Shuzhe Wang, Vincent Leroy, Yohann Cabon, Boris Chidlovskii, and Jerome Revaud. Dust3r: Geometric 3d vision made easy. In *CVPR*, 2024.

Haoyu Wu, Diankun Wu, Tianyu He, Junliang Guo, Yang Ye, Yueqi Duan, and Jiang Bian. Geometry forcing: Marrying video diffusion and 3d representation for consistent world modeling. In *ICLR*, 2026.

Chengen Xie, Bin Sun, Tianyu Li, Junjie Wu, Zhihui Hao, XianPeng Lang, and Hongyang Li. Latentvla: Efficient vision-language models for autonomous driving via latent action prediction. *arXiv preprint arXiv:2601.05611*, 2026.

Zebin Xing, Xingyu Zhang, Yang Hu, Bo Jiang, Tong He, Qian Zhang, Xiaoxiao Long, and Wei Yin. Goalflow: Goal-driven flow matching for multimodal trajectories generation in end-to-end autonomous driving. In *CVPR*, 2025.

Zhiyuan Xu, Bohan Li, Huan-ang Gao, Mingju Gao, Yong Chen, Ming Liu, Chenxu Yan, Hang Zhao, Shuo Feng, and Hao Zhao. Challenger: Affordable adversarial driving video generation. *arXiv preprint arXiv:2505.15880*, 2025.

Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *CVPR*, 2024a.

Lihe Yang, Bingyi Kang, Zilong Huang, Zhen Zhao, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything v2. In *NeurIPS*, 2024b.

Ziyang Yao, Haochen Liu, Yuncheng Jiang, Zeyu Zhu, Zibin Guo, Jingru Wang, Tianle Liu, Jianwei Cui, Kuiyuan Yang, Hongwei Xie, et al. Discrete-wam: Unified discrete vision-action token editing for world-policy learning, 2026.

Rui Yu, Xianghang Zhang, Runkai Zhao, Huaicheng Yan, and Meng Wang. Distilldrive: End-to-end multi-mode autonomous driving distillation by isomorphic hetero-source planning model. In *ICCV*, 2025a.

Sihyun Yu, Sangkyung Kwak, Huiwon Jang, Jongheon Jeong, Jonathan Huang, Jinwoo Shin, and Saining Xie. Representation alignment for generation: Training diffusion transformers is easier than you think. In *ICLR*, 2025b.

Sergey Zagoruyko and Nikos Komodakis. Paying more attention to attention: Improving the performance of convolutional neural networks via attention transfer. *arXiv preprint arXiv:1612.03928*, 2016.

Kaiwen Zhang, Zhenyu Tang, Xiaotao Hu, Xingang Pan, Xiaoyang Guo, Yuan Liu, Jingwei Huang, Li Yuan, Qian Zhang, Xiao-Xiao Long, et al. Epona: Autoregressive diffusion world model for autonomous driving. In *ICCV*, 2025.

Yinan Zheng, Tianyi Tan, Bin Huang, Enguang Liu, Ruiming Liang, Jianlin Zhang, Jianwei Cui, et al. Unleashing the potential of diffusion models for end-to-end autonomous driving. *arXiv preprint arXiv:2602.22801*, 2026.

Yupeng Zheng, Pengxuan Yang, Zebin Xing, Qichao Zhang, Yuhang Zheng, Yinfeng Gao, Pengfei Li, Teng Zhang, Zhongpu Xia, Peng Jia, et al. World4drive: End-to-end autonomous driving via intention-aware physical latent world model. In *ICCV*, 2025.

Hongyu Zhou, Longzhong Lin, Jiabao Wang, Yichong Lu, Dongfeng Bai, Bingbing Liu, Yue Wang, Andreas Geiger, and Yiyi Liao. Hugsim: A real-time, photo-realistic and closed-loop simulator for autonomous driving. *IEEE TPAMI*, 2025a.

Yunsong Zhou, Naisheng Ye, William Ljungbergh, Tianyu Li, Jiazhi Yang, Zetong Yang, Hongzi Zhu, Christoffer Petersson, and Hongyang Li. Decoupled diffusion sparks adaptive scene generation. In *ICCV*, 2025b.

Jialv Zou, Shaoyu Chen, Bencheng Liao, Zhiyu Zheng, Yuehao Song, Lefei Zhang, Qian Zhang, Wenyu Liu, and Xinggang Wang. Diffusiondrive-v2: Reinforcement learning-constrained truncated diffusion modeling in end-to-end autonomous driving, 2025.

Sicheng Zuo, Zixun Xie, Wenzhao Zheng, Shaoqing Xu, Fang Li, Shengyin Jiang, Long Chen, Zhi-Xin Yang, and Jiwen Lu. Dvgt: Driving visual geometry transformer. In *CVPR*, 2026a.

Sicheng Zuo, Zixun Xie, Wenzhao Zheng, Shaoqing Xu, Fang Li, Hanbing Li, Long Chen, Zhi-Xin Yang, and Jiwen Lu. Dvgt-2: Vision-geometry-action model for autonomous driving at scale. *arXiv preprint arXiv:2604.00813*, 2026b.

Appendix

The appendix is organized as follows. Appendix A provides the method details. Appendix B provides evaluation and implementation details. Appendix C reports additional quantitative results. Appendix D presents additional qualitative visualizations and failure cases. Appendix E discusses limitations and broader impact. Appendix F lists the licenses of the assets used in this work.

A VFM-GUIDED REPRESENTATION ALIGNMENT

We present the ViRA framework as follows. Section 2 defines the notation and problem setup. Appendix A.1 provides an overview of the ViRA framework for E2E planners. Appendix A.2 introduces Spatial Dependency Alignment (SDA) and Scene Semantic Alignment (SSA). Appendix A.3 summarizes the training objective.

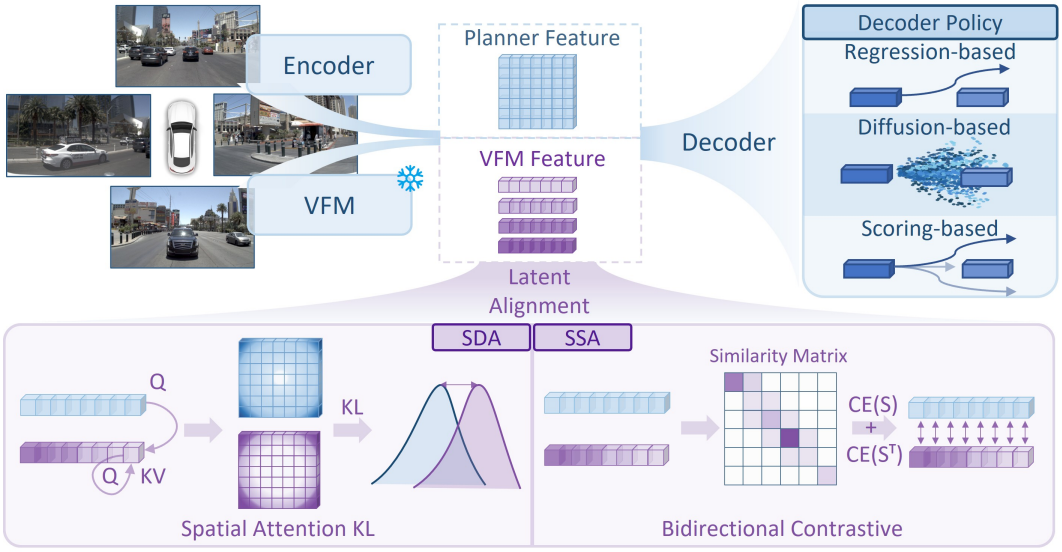

Figure S.1: **Pipeline of ViRA.** Multi-view images are encoded by both the planner encoder and a frozen VFM target, producing student and VFM representations. The VFM representations supervise latent alignment only, without being injected into the policy head. The policy head operates solely on planner representations, and the VFM branch is removed after training, yielding no inference overhead. Q, K, and V denote query, key, and value; KL denotes Kullback–Leibler divergence; CE denotes cross-entropy.

A.1 VFM-GUIDED E2E PLANNER

As illustrated in Figure S.1, given a single-frame multi-view observation $\mathcal{O}$, a conventional E2E planner first extracts an intermediate representation $\mathbf{F}^s$ using the planner encoder E_θ, and then predicts the future trajectory $\hat{\tau}$ with a driving policy π_ψ:

$$\mathbf{F}^s = E_\theta(\mathcal{O}), \qquad \hat{\tau} = \pi_\psi(\mathbf{F}^s). \tag{S.1}$$

Here, π_ψ denotes a general policy head, which can be instantiated by different E2E planner designs, such as regression-based (Chitta et al., 2023), diffusion-based (Liao et al., 2025), or scoring-based (Feng et al., 2026) policies.

To improve the representation quality of the planner encoder, we additionally feed the same observation $\mathcal{O}$ into a pre-trained VFM T_ϕ to obtain a VFM representation $\mathbf{F}^t = T_\phi(\mathcal{O})$, where ϕ is frozen throughout training. The VFM representation is used only to compute the representation alignment loss and is then discarded, without being passed to the policy head. Unless otherwise specified, we use DVGT (Zuo et al., 2026a) as the default VFM target T_ϕ, a VFM pre-trained on autonomous driving data, while our framework is also compatible with other VFMs (Carion et al., 2025; Wang

et al., 2025; Siméoni et al., 2025; Lin et al., 2025). Importantly, the VFM and alignment heads are used only during training and discarded during inference, keeping the original planner architecture and inference pipeline unchanged with no additional inference cost. Since ViRA uses frozen VFM representations as supervision, it requires no extra annotations.

A.2 Latent Representation Alignment

To transfer driving-relevant knowledge from VFMs to planner encoders, we align their representations from two complementary perspectives: spatial distribution and scene semantic structure. As shown in Figure S.1, Spatial Dependency Alignment (SDA) aligns the spatial attention distribution of the planner representation with that of the VFM representation, encouraging the planner to capture fine-grained spatial dependencies. Scene Semantic Alignment (SSA) further aligns the global latent representations of the E2E planner and the VFMs through a bidirectional contrastive objective, preserving scene-level semantic consistency across training samples. Together, these two objectives improve the representation quality of the E2E planner encoder for downstream trajectory planning.

Spatial Dependency Alignment. SDA aims to transfer the spatial relational structure encoded by the VFM to the E2E planner representation. Different from directly matching feature values at each spatial location, we align the spatial dependency distributions between planner and VFM features, following the spirit of attention-based knowledge distillation (Zagoruyko & Komodakis, 2016; Tung & Mori, 2019). Given projected and flattened planner and VFM features, we define the SDA loss as

$$\mathcal{L}_{\text{SDA}} = \mathbb{E}_{b,i}\left[D_{\text{KL}}\left(\mathbf{A}^t_{b,i,:} \,\|\, \mathbf{A}^s_{b,i,:}\right)\right], \tag{S.2}$$

where the expectation is taken over all samples in the batch and all spatial tokens. Here, D_{KL} denotes the Kullback–Leibler divergence, $\mathbf{A}^s = \text{softmax}(\mathbf{Q}^s(\mathbf{K}^t)^\top/\gamma_{sda})$ and $\mathbf{A}^t = \text{softmax}(\mathbf{Q}^t(\mathbf{K}^t)^\top/\gamma_{sda})$ denote the planner-to-VFM dependency distribution and the VFM self-dependency distribution, respectively. $\mathbf{Q}^s$ and $\mathbf{Q}^t$ are query projections of the planner and VFM features, and $\mathbf{K}^t$ is the key projection of the VFM features. We set the SDA temperature to $\gamma_{\text{sda}} = 0.07$. This objective encourages the planner encoder to reproduce the spatial dependencies distilled from the frozen VFM.

Scene Semantic Alignment. SSA encourages the planner representation to preserve global semantic consistency with the VFM, following the standard contrastive learning formulation (Oord et al., 2018; Chen et al., 2020; Radford et al., 2021). After aggregating the projected planner and VFM features into global embeddings $\mathbf{z}^s$ and $\mathbf{z}^t$, we apply a bidirectional contrastive loss:

$$\mathcal{L}_{\text{SSA}} = \frac{1}{2}\left[\text{CE}(\mathbf{M}, \mathbf{y}) + \text{CE}(\mathbf{M}^\top, \mathbf{y})\right], \tag{S.3}$$

where CE denotes the cross-entropy loss, $\mathbf{M} = \mathbf{z}^s(\mathbf{z}^t)^\top/\gamma_{\text{ssa}}$ is the planner–VFM similarity matrix, $\mathbf{y} = [1, \ldots, B]$ denotes the matched scene indices within the batch, with B being the batch size. And we set the SSA temperature to $\gamma_{\text{ssa}} = 1.0$. The row-wise and column-wise cross-entropy terms enforce planner-to-VFM and VFM-to-planner matching, respectively.

A.3 Training Objective

Different E2E planners use different training objectives depending on their architectural designs. Some are trained with trajectory supervision only, while others include auxiliary perception objectives, such as detection, segmentation, or depth estimation. ViRA is compatible with both settings: we keep the original E2E planner architecture and training protocol unchanged, and simply add representation alignment losses as representation-level supervision. Let $\mathcal{L}_{\text{planner}}$ denote the original training objective of a planner, including its trajectory loss and any native auxiliary supervision if used. The overall training objective is:

$$\mathcal{L} = \mathcal{L}_{\text{planner}} + \lambda_{\text{SDA}}\mathcal{L}_{\text{SDA}} + \lambda_{\text{SSA}}\mathcal{L}_{\text{SSA}}. \tag{S.4}$$

Here, $\mathcal{L}_{\text{SDA}}$ and $\mathcal{L}_{\text{SSA}}$ denote the proposed spatial and semantic latent alignment losses, respectively. Motivated by recent findings that spatial structure is more critical than global semantics in representation alignment (Singh et al., 2026), we set $\lambda_{\text{SDA}} = 1$ and $\lambda_{\text{SSA}} = 0.1$. For planning-only planners, $\mathcal{L}_{\text{planner}}$ contains only the trajectory prediction objective. For planners with auxiliary objectives, the original auxiliary supervision terms are included in $\mathcal{L}_{\text{planner}}$ and kept unchanged.

B Implementation and Evaluation Details

B.1 Benchmark and Metrics

NAVSIM. We evaluate driving performance on navtest and navhard using NAVSIM v2 (Cao et al., 2025). The navtest evaluation covers 12,146 real-world scenarios in a single stage, providing a broad assessment across driving conditions. For navhard, we follow the same two-stage evaluation setup as SimScale (Tian et al., 2026), using 244 challenging real-world scenarios and 4,164 synthetic follow-up scenarios rendered with 3D Gaussian Splatting. Both evaluations use the Extended Predictive Driver Model Score (EPDMS) (Li et al., 2025a; Cao et al., 2025), which combines multiplicative penalties with a weighted average of driving-quality scores.

$$\text{EPDMS} = \underbrace{\prod_{m \in \mathcal{M}_{\text{pen}}} S_m}_{\text{penalties}} \cdot \underbrace{\frac{\sum_{m \in \mathcal{M}_{\text{avg}}} w_m S_m}{\sum_{m \in \mathcal{M}_{\text{avg}}} w_m}}_{\text{weighted average}}, \tag{S.5}$$

where S_m denotes the evaluated subscore for metric m, including human-reference filtering where enabled, and w_m is its weight. The penalty set $\mathcal{M}_{\text{pen}}$ comprises No-at-fault Collisions (NC), Drivable Area Compliance (DAC), Driving Direction Compliance (DDC), and Traffic Light Compliance (TLC). The weighted set $\mathcal{M}_{\text{avg}}$ comprises Time-to-Collision (TTC), Ego Progress (EP), Lane Keeping (LK), History Comfort (HC), and Extended Comfort (EC). Human-reference filtering assigns a subscore of one when the corresponding human-reference score is zero. For navhard, evaluation uses reactive traffic agents and combines scores from the initial and synthetic follow-up scenarios through two-stage aggregation.

HUGSIM. HUGSIM (Zhou et al., 2025a) is a real-time, photorealistic closed-loop simulator that reconstructs 3D scenes from captured RGB images with 3D Gaussian Splatting and updates ego and actor states at each step from the planner output, so that surrounding vehicles react to the ego. We evaluate planners trained only on NAVSIM in a zero-shot manner on scenarios grouped into four difficulty levels: Easy, Medium, Hard, and Extreme. Each episode is scored by HD-Score (Zhou et al., 2025a), reported on a 0–100 scale per level and on average:

$$\text{HD-Score} = \text{RC} \times \frac{1}{T} \sum_{t=1}^{T} \left(\text{NC}_t \times \text{DAC}_t \times \frac{5 \cdot \text{TTC}_t + 2 \cdot \text{Comf}_t}{7} \right), \tag{S.6}$$

where RC is the fraction of the intended route completed within an episode of T steps, NC and DAC are defined as above, and TTC and Comfort (Comf.) take values in $[0, 1]$.

mIoU. We measure the semantics encoded in planner representations with the BEV segmentation head of TransFuser. Mean Intersection-over-Union (mIoU) is the unweighted average of per-class IoU over seven classes: background, road, walkways, centerline, static objects, vehicles, and pedestrians.

$$\text{IoU}_c = \frac{|P_c \cap G_c|}{|P_c \cup G_c|}, \quad \text{mIoU} = \frac{1}{C} \sum_{c=1}^{C} \text{IoU}_c, \tag{S.7}$$

where $C = 7$, P_c is the set of BEV cells predicted as class c over the evaluation set, and G_c is the corresponding ground-truth set. The main-text table reports class-wise IoU only for background, road, walkways, centerline, and vehicles. Static objects and pedestrians remain below 0.5 IoU for every evaluated target, so they are omitted from that table; the reported mIoU still averages all seven classes. Table S.6 lists the score of every class.

B.2 Training Details

This section complements the experimental setup in Section 2.2 by providing the model configurations and training hyperparameters for all evaluated planners. As summarized in Table S.1, we follow the NAVSIM (Dauner et al., 2024) training protocol unless otherwise specified, and train all models on the training logs (Karnchanachari et al., 2024) of navtrain. Each planner is trained with its corresponding official settings, while ViRA leaves the planner architecture and inference pipeline unchanged.

TransFuser (Chitta et al., 2023) is a regression-based planner built on fused sensory representations. It uses three camera views with a resolution of 2048×512, predicts a 4s future trajectory at 2Hz, and adopts ResNet-34 (He et al., 2016) as the image backbone. The model contains 56M parameters and uses detection and segmentation as auxiliary tasks. We train TransFuser for 100 epochs on 8 H100 GPUs with a total batch size of 192, an initial learning rate of 1×10^{-4}, a constant learning-rate schedule, and the Adam optimizer (Kingma & Ba, 2015).

DiffusionDrive (Liao et al., 2025) is a generative planner that predicts future trajectories through diffusion-based trajectory modeling. It uses the same three-camera input resolution, prediction horizon, and frequency as TransFuser, i.e., 2048×512 images, a 4s horizon, and 2Hz trajectory frequency. It also adopts ResNet-34 (He et al., 2016) as the image backbone and uses detection and segmentation auxiliary tasks. The model contains 61M parameters. We train DiffusionDrive for 100 epochs on 8 H100 GPUs with a total batch size of 192, an initial learning rate of 6×10^{-4}, a cosine decay schedule, and the AdamW optimizer (Loshchilov & Hutter, 2019).

Rap* (Feng et al., 2026) is a scoring-based planner that selects trajectories from a predefined trajectory vocabulary. It takes four camera views with a resolution of $4 \times 768 \times 448$, predicts a 5s future trajectory at 2Hz, and uses ResNet-34 (He et al., 2016) as the image backbone. The corresponding model size is 69M. Unlike TransFuser and DiffusionDrive, Rap* does not use auxiliary detection or segmentation tasks. We train Rap* for 30 epochs on 8 H100 GPUs with a total batch size of 48, an initial learning rate of 1×10^{-4}, a cosine decay schedule, and the Adam optimizer (Kingma & Ba, 2015).

Table S.1: **Model and training hyperparameters.**

Hyperpara.	**TransFuser** (Chitta et al., 2023)	**DiffusionDrive** (Liao et al., 2025)	**Rap*** (Feng et al., 2026)
	Model Configuration		
Sensors	$3 \times$ Cam.	$3 \times$ Cam.	$4 \times$ Cam.
Resolution	2048×512	2048×512	$4 \times 768 \times 448$
Horizon	4s	4s	5s
Frequency	2Hz	2Hz	2Hz
Backbone	R34 (He et al., 2016)	R34 (He et al., 2016)	R34 (He et al., 2016)
Parameters	56M	61M	69M
Aux. Tasks	Det. Seg.	Det. Seg.	None
	Training Configuration		
GPUs	$8 \times$ H100	$8 \times$ H100	$8 \times$ H100
Epochs	100	100	30
Total BS	192	192	48
Initial LR	1×10^{-4}	6×10^{-4}	1×10^{-4}
Schedule	Constant	Cosine Decay	Cosine Decay
Optimizer	Adam (Kingma & Ba, 2015)	AdamW (Loshchilov & Hutter, 2019)	Adam (Kingma & Ba, 2015)

B.3 Latent Alignment Implementation

We provide additional implementation details of the representation alignment module. In our experiments, we use the official public checkpoints of DVGT (Zuo et al., 2026a), VGGT (Wang et al., 2025), DINOv3 (Siméoni et al., 2025), SAM3 (Carion et al., 2025), and DA3 (Lin et al., 2025) as frozen VFM targets, and keep all VFM parameters frozen throughout training. For all VFM targets, we use the intermediate representations before their task-specific prediction heads, rather than the final task outputs. This design avoids relying on explicit pseudo labels and instead transfers the VFM's learned visual representations to the planner encoder.

For DVGT (Zuo et al., 2026a), we follow its original feature extraction design and take the multi-level features before the DPT prediction head. Specifically, we extract four intermediate token features from the DVGT aggregator, fuse them with learnable weights, and reshape the patch tokens into a spatial feature map. For other VFM targets, we follow the same principle and use the corresponding feature maps before their prediction heads.

Given the planner feature $\mathbf{F}^s$ and the VFM feature $\mathbf{F}^t$, we first resize $\mathbf{F}^s$ to the spatial resolution of $\mathbf{F}^t$. Both features are then projected into a shared latent space by lightweight 1×1 convolutional

Table S.2: **Effect of alignment objectives on navtest.** MSE, Cosine, KL, and Contrastive denote mean squared error, cosine similarity, Kullback–Leibler divergence, and contrastive loss, respectively.

MSE	Cosine	KL	Contrastive	DAC	DDC	TLC	LK	EPDMS ↑
✓	✗	✗	✗	90.9	98.6	99.8	92.3	84.4
✗	✓	✗	✗	90.8	99.0	99.8	95.6	85.1
✗	✗	✓	✗	90.3	98.9	99.7	94.6	84.3
✗	✗	✗	✓	90.4	98.9	99.8	95.8	84.6
✓	✓	✗	✗	89.6	99.0	99.7	95.9	83.6
✓	✗	✓	✗	92.9	99.4	99.8	95.8	87.8
✗	✓	✗	✓	93.5	99.4	99.8	96.4	88.1
✗	✗	✓	✓	94.1	99.5	99.8	96.8	**89.0**

projection heads. The projected features are used to compute the spatial distribution alignment loss and the semantic consistency loss described in Appendix A.2. The VFM and alignment heads are used only during training and discarded after training, introducing no additional inference cost.

C Additional Quantitative Experiments

C.1 Different Alignment Objectives Ablation

To validate the effectiveness of the latent alignment objective, we conduct an ablation study on TransFuser (Chitta et al., 2023) with different alignment objectives. We consider representative losses for feature-level, distribution-level, and contrastive supervision, and compare different combinations for spatial and semantic alignment (Yu et al., 2025b; Li et al., 2026; Wu et al., 2026). As shown in Table S.2, jointly aligning spatial dependencies and semantic consistency generally yields better planning performance than using a single alignment objective. Among all combinations, KL divergence for spatial alignment and contrastive loss for semantic alignment achieve the best performance, reaching an EPDMS of 89.0.

C.2 Supplementary Alignment Analysis

To further verify the alignment effectiveness of ViRA, we compute the Pearson correlation between planner encoder features and VFM features. As shown in Figure S.2, the original TransFuser encoder exhibits weak correlation with the VFM features, and the resulting correlation matrix lacks clear structural patterns. After DVGT supervision, TransFuser with DVGT shows stronger and more structured correlation patterns, indicating that ViRA effectively narrows the representation gap between the planner encoder and the VFM. This result provides feature-level evidence for the effectiveness of ViRA.

C.3 VFM Target Dependency Analysis

This section provides additional quantitative results for the VFM-target analysis. We evaluate ViRA on both TransFuser (Chitta et al., 2023) and Rap* (Feng et al., 2026). TransFuser uses detection and segmentation as perception auxiliary tasks, whereas Rap* does not rely on explicit perception supervision, allowing us to examine whether ViRA behaves consistently across different planner designs.

VFM-target dependency across planners. Across the five targets used in the main comparison, ViRA improves TransFuser by 5.2–5.7 EPDMS and Rap* by 6.6–12.1 EPDMS (Tables S.3 and S.5). Auxiliary supervision therefore also narrows the spread in this expanded target set.

Random-target control. To test whether the gains depend on VFM pre-training, we replace the pre-trained target with a randomly initialized VFM target while retaining the same alignment objectives. As shown in Tables S.3 and S.5, random-target alignment lowers EPDMS by 1.6 points for TransFuser and 1.7 points for Rap*. Pre-trained DVGT improves both planners, by $+5.4$ and $+10.9$ points,

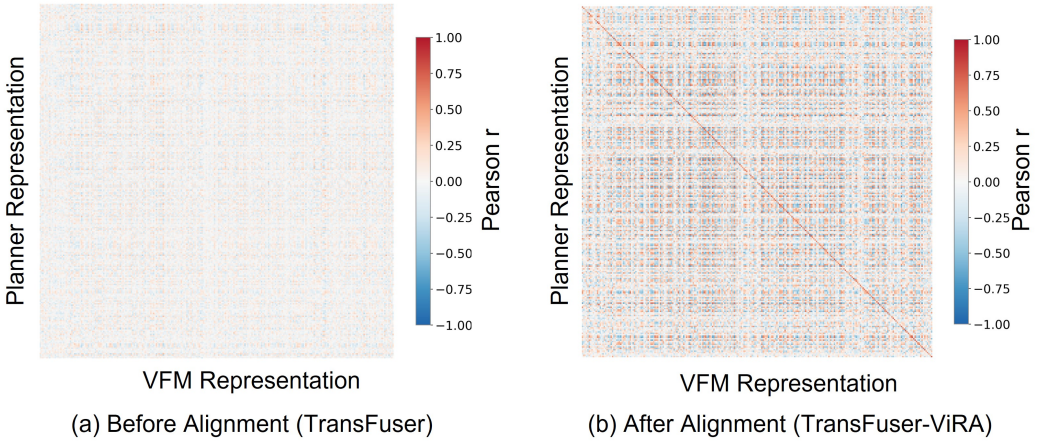

Figure S.2: **Feature correlation analysis.**

Table S.3: **Effect of different VFM teachers on TransFuser.** Performance on the NAVSIM v2 navtest benchmark.

Teachers	NC ↑	DAC ↑	DDC ↑	TLC ↑	EP ↑	TTC ↑	LK ↑	HC ↑	EC ↑	EPDMS ↑
Baseline	97.1	89.6	99.0	99.8	97.9	95.9	95.9	98.3	87.3	83.6
Random init	96.8	90.2	99.0	99.7	98.0	96.1	95.5	98.3	86.7	82.0 -1.6
DVGT (Zuo et al., 2026a)	97.7	94.1	99.5	99.8	98.2	96.7	96.8	98.3	87.9	89.0 +5.4
VGGT (Wang et al., 2025)	97.8	94.6	99.3	99.8	97.8	97.0	96.7	98.3	87.2	89.3 +5.7
DA3 (Lin et al., 2025)	97.9	94.2	99.5	99.8	98.0	97.0	95.9	98.3	87.5	89.1 +5.5
DINOv3-L (Siméoni et al., 2025)	97.6	94.4	99.3	99.7	98.0	97.0	96.7	98.3	86.6	88.8 +5.2
SAM3 (Carion et al., 2025)	97.9	94.2	99.5	99.8	98.0	96.9	95.1	98.3	88.1	89.0 +5.4

Table S.4: **Effect of different VFM teachers on TransFuser without auxiliary perception supervision.** Performance on the NAVSIM v2 navtest benchmark.

Teachers	NC ↑	DAC ↑	DDC ↑	TLC ↑	EP ↑	TTC ↑	LK ↑	HC ↑	EC ↑	EPDMS ↑
Baseline	97.0	89.9	98.5	99.8	97.9	96.3	94.3	98.4	86.8	82.6
Random init	95.4	90.9	97.7	99.5	98.1	94.2	93.6	98.3	84.8	80.1 -2.5
DVGT (Zuo et al., 2026a)	97.9	90.9	99.1	99.7	98.0	96.8	96.5	98.3	87.2	88.2 +5.6
VGGT (Wang et al., 2025)	97.7	92.9	99.1	99.8	98.0	96.6	95.9	98.3	87.5	87.6 +5.0
DA3 (Lin et al., 2025)	97.9	92.4	99.1	99.7	97.9	96.7	95.9	98.3	85.8	87.2 +4.6
DINOv3-L (Siméoni et al., 2025)	97.9	90.9	99.3	99.8	98.2	96.9	96.7	98.5	88.6	89.1 +6.5
SAM3 (Carion et al., 2025)	97.6	91.5	99.0	99.8	97.9	96.4	96.0	98.4	88.2	86.4 +3.8

respectively. Applying the same alignment objectives to random features therefore does not reproduce the gains observed with pre-trained VFM representations.

BEV semantic map analysis. We further report the BEV semantic map results of TransFuser in Table S.6. All pre-trained VFM targets improve BEV segmentation mIoU, but the gains vary from +0.5 (SAM3) to +3.0 (DINOv3).

D Additional Qualitative Visualizations

D.1 Additional Qualitative Analysis

We provide additional qualitative visualizations to further analyze the effect of ViRA across different planners, VFM targets, and driving scenarios. These examples include perception-based planners in

Table S.5: **Effect of different VFM teachers on Rap*.** Performance on the NAVSIM v2 navtest benchmark. Rap* is our reimplementation with a different backbone from the official codebase and without the original data augmentation.

Teachers	NC ↑	DAC ↑	DDC ↑	TLC ↑	EP ↑	TTC ↑	LK ↑	HC ↑	EC ↑	EPDMS ↑
Baseline	91.6	88.8	93.5	99.8	87.2	92.5	90.9	97.2	56.7	72.8
Random init	94.6	88.1	95.1	99.8	84.2	95.1	93.1	96.7	41.9	71.1 -1.7
DVGT (Zuo et al., 2026a)	97.1	96.9	99.2	99.9	87.5	96.9	96.2	97.7	48.1	83.7 +10.9
VGGT (Wang et al., 2025)	96.4	95.6	98.6	99.9	84.0	96.9	95.1	97.4	36.5	79.4 +6.6
DA3 (Lin et al., 2025)	96.1	95.6	98.6	99.7	88.8	96.1	96.0	97.7	54.9	82.5 +9.7
DINOv3 (Siméoni et al., 2025)	97.6	96.4	98.9	100.0	87.8	97.7	95.9	98.0	58.4	84.9 +12.1
SAM3 (Carion et al., 2025)	97.1	95.5	98.4	99.8	89.9	96.9	95.5	96.8	60.1	84.0 +11.2

Table S.6: **VFMs dependency on BEV segmentation of TransFuser.** Class-wise BEV segmentation performance of TransFuser under different VFM targets. BG indicates background, SO indicates static objects.

Teachers	BG ↑	Road ↑	Walkways ↑	Centerline ↑	SO ↑	Vehicles ↑	Pedestrians ↑	mIoU ↑
Baseline	85.4	62.3	56.3	23.6	0.1	27.1	0.0	36.4
DA3 (Lin et al., 2025)	86.6	63.8	59.1	28.2	0.3	32.3	0.0	38.6 +2.2
VGGT (Wang et al., 2025)	86.5	63.9	58.5	27.6	0.3	33.6	0.0	38.6 +2.2
SAM3 (Carion et al., 2025)	85.8	62.4	56.7	25.0	0.1	28.0	0.0	36.9 +0.5
DVGT (Zuo et al., 2026a)	86.9	64.6	59.3	28.9	0.5	33.4	0.2	39.1 +2.7
DINOv3 (Siméoni et al., 2025)	87.1	64.7	60.1	29.8	0.3	33.5	0.2	39.4 +3.0

complex intersection scenes Figure S.3, and comparisons under different VFM targets Figure S.4. Together, these visualizations complement the quantitative results and show that ViRA consistently improves scene understanding and planning behavior across different settings.

D.2 Failure Cases

Although ViRA improves representation quality and downstream planning performance, challenging failure cases remain. For perception-free planners Figure S.5, different VFM targets exhibit slightly different failure patterns, suggesting that their representations may differ in subtle but meaningful ways and lead to small performance variations. For perception-based planners Figure S.6, even when ViRA produces more complete BEV semantic maps, the planned trajectory may still fail to follow the human expert in challenging scenarios. These cases show that high-quality visual representations are fundamental, but not a silver bullet for end-to-end autonomous driving. Robust planning still requires better trajectory decoding, temporal reasoning, interaction modeling, and decision-making under uncertainty.

E Limitations and Broader Impact

Training-time VFM Cost. Although ViRA introduces no additional inference overhead, it relies on external frozen VFM representations during training. Extracting and storing VFM representations may introduce extra training cost, especially when using large-scale VFMs or high-resolution multi-view inputs. A practical future direction is to reduce this cost through feature caching, lightweight VFM distillation, or selective alignment over only the most informative layers and spatial regions. This would make VFM-guided representation alignment more scalable for larger datasets and more diverse planner architectures.

Scope of Architectures and Benchmarks. This work evaluates ViRA on three representative E2E planners, on NAVSIM v2 navtest and navhard, and in zero-shot closed-loop evaluation on HUGSIM. It does not cover multi-modal sensor fusion, vision-language-action models, or real-vehicle deployment.

Further validation on a wider set of planners and real-world data is needed to assess how far the observed planner-dependent conversion from representation quality to planning gains generalizes.

Broader Impact. By improving the visual representations used by E2E planners, ViRA may contribute to safer and more reliable autonomous driving systems. Meanwhile, the use of large-scale VFMs also raises potential concerns. VFMs may inherit biases from their pre-training data, including geographical, weather, cultural, and traffic-distribution biases. If such biases are transferred to autonomous driving planners without careful validation, they may cause degraded performance in underrepresented regions or rare safety-critical scenarios. Therefore, we believe that VFM-guided planning should be accompanied by rigorous evaluation across diverse environments, transparent failure analysis, and safety-oriented validation before real-world deployment.

F License of Assets

All training and evaluation are conducted on publicly available datasets and benchmarks, including nuPlan (Karnchanachari et al., 2024), OpenScene (OpenScene Contributors, 2023), and NAVSIM v2 (Cao et al., 2025). We use these datasets only for research purposes and follow their official licenses and terms of use. We adopt several publicly available VFMs as VFM targets, including DVGT (Zuo et al., 2026a), VGGT (Wang et al., 2025), DINOv3 (Siméoni et al., 2025), SAM3 (Carion et al., 2025), and DA3 (Lin et al., 2025). DVGT is released under the Apache License 2.0. VGGT, DINOv3, and SAM3 are used under their respective official licenses. For DA3, we follow the license associated with the specific released model checkpoint used in our experiments. We also build upon publicly available E2E planners, including Rap* (Feng et al., 2026), DiffusionDrive (Liao et al., 2025), and TransFuser (Chitta et al., 2023). Rap* is released under the Apache License 2.0, while DiffusionDrive and TransFuser are released under the MIT License. We do not redistribute third-party datasets, model weights, or checkpoints. All third-party assets are used in accordance with their corresponding licenses. Our released code will include proper attribution and license notices for all used assets.

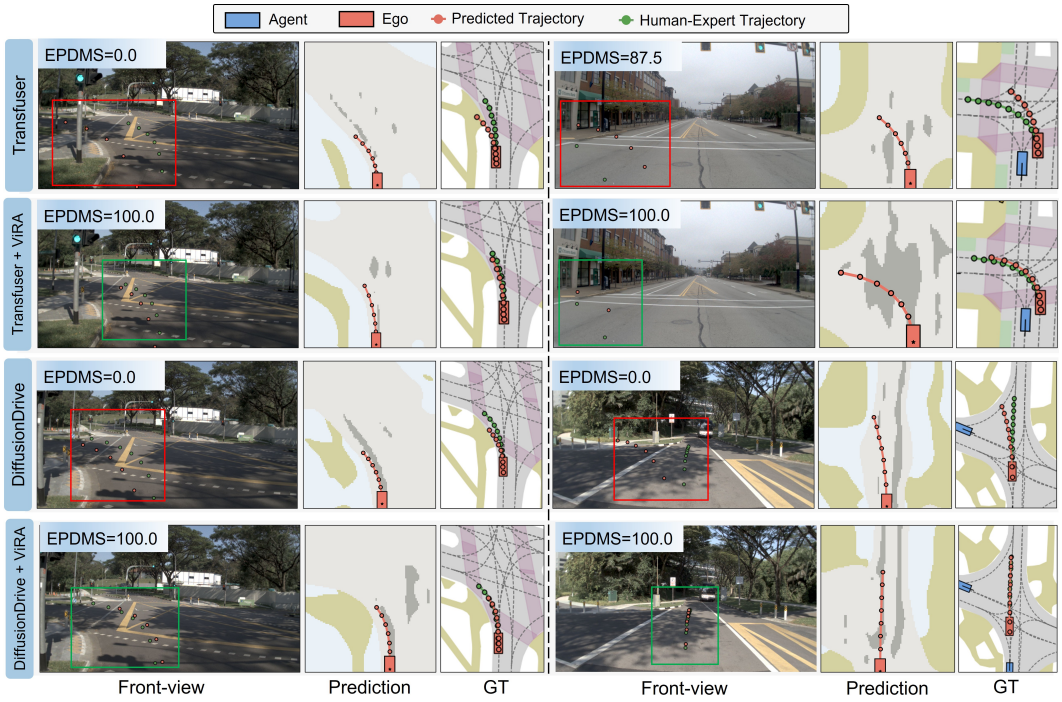

Figure S.3: **Additional qualitative analysis for perception-based planners.**

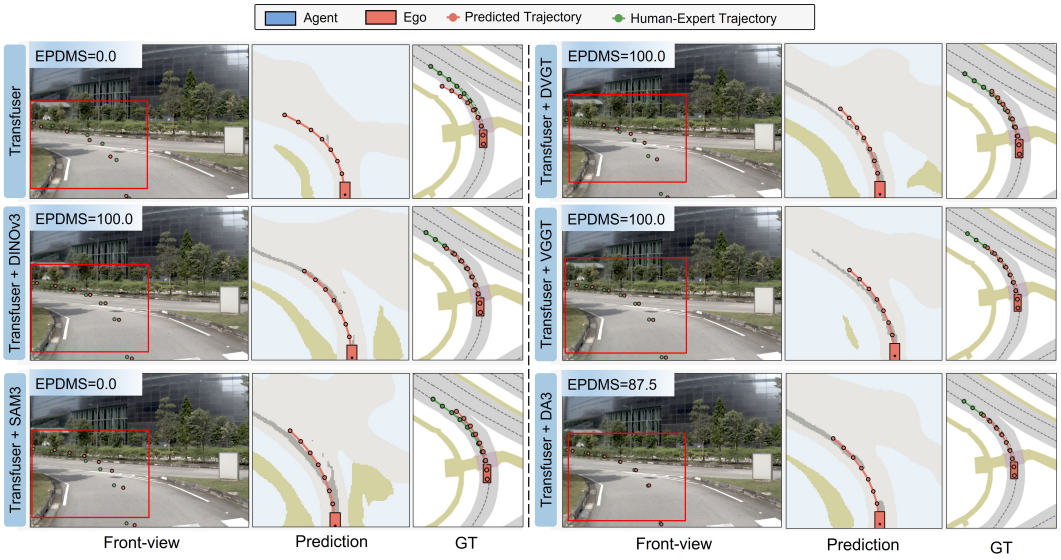

Figure S.4: **Additional qualitative analysis for perception-based planners across different VFMs.**

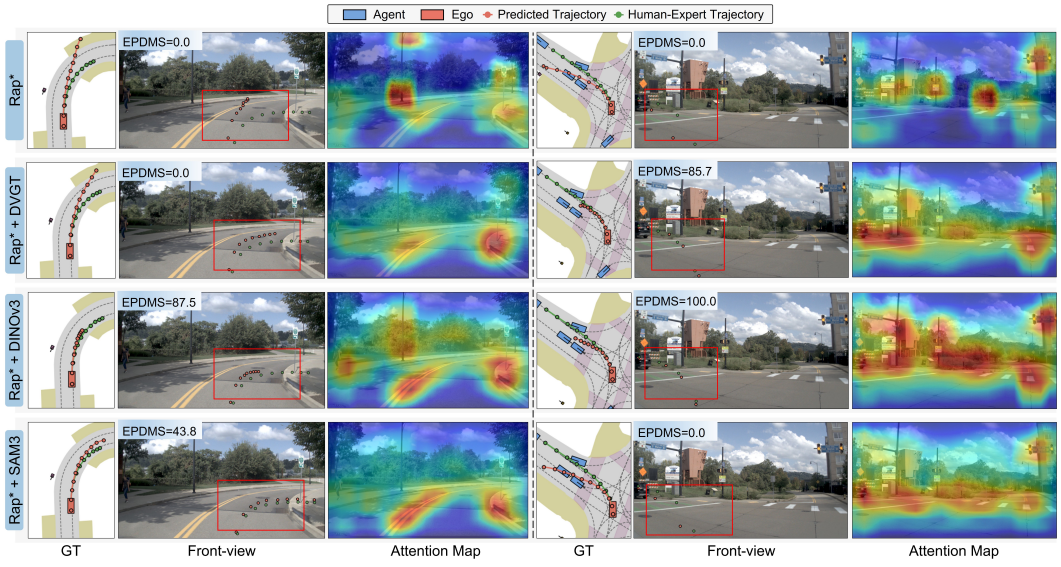

Figure S.5: **Failure case analysis for a perception-free planner.**

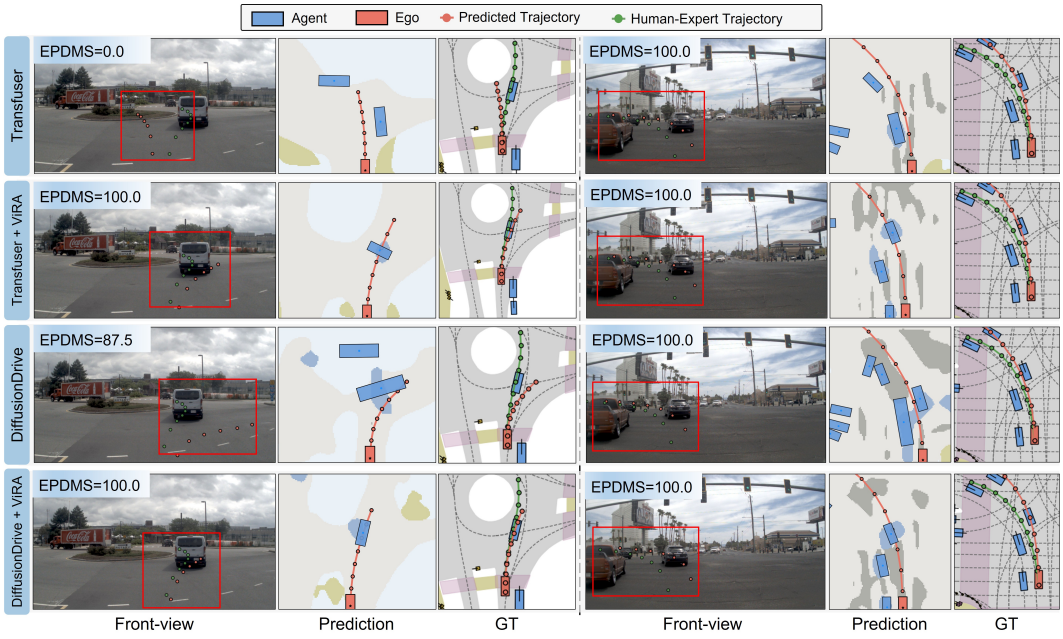

Figure S.6: **Failure case analysis for perception-based planners.**