DRIVE-KD: MULTI-TEACHER DISTILLATION FOR VLMS IN AUTONOMOUS DRIVING

Anonymous authors
Paper under double-blind review

ABSTRACT

Autonomous driving is an important and safety-critical task, and recent advances in LLMs/VLMs have opened new possibilities for reasoning and planning in this domain. However, large models demand substantial GPU memory and exhibit high inference latency, while conventional supervised fine-tuning (SFT) often struggles to bridge the capability gaps of small models. To address these limitations, we propose Drive-KD, a framework that decomposes autonomous driving into a "perception–reasoning–planning" triad and transfers these capabilities via knowledge distillation to improve VLM performance on comprehensive driving tasks. Through a systematic study of distillation design for autonomous driving, we identify layer-specific attention as the distillation signal to construct capability-specific single-teacher models. Moreover, we unify these single-teacher settings into a multi-teacher distillation framework and introduce asymmetric gradient projection to mitigate cross-capability gradient conflicts. Experiments show that our distilled InternVL3-1B model, with $\sim$42$\times$ less GPU memory and $\sim$11.4$\times$ higher throughput, achieves better overall performance than the pretrained 78B model from the same family on DriveBench, and surpasses GPT-5.1 on the planning dimension. It further substantially outperforms its SFT counterpart in closed-loop evaluation on Bench2Drive-VL, achieving 84.50 Route Completion. Further attention analysis shows that Drive-KD restores the capability-specific visual grounding suppressed by SFT.

1 INTRODUCTION

Autonomous driving is a complex domain with high safety requirements Hu et al. (2023); Zhao et al. (2024). Classical autonomous driving solutions mainly follow two paradigms Fourati et al. (2026). One is a modular pipeline that decomposes object detection, tracking, behavior prediction, and trajectory planning into separate submodules Hu et al. (2025); Badue et al. (2021), and the other is an end-to-end learning paradigm that directly maps sensor inputs to control commands or semantic representations Bojarski et al. (2016); Jiang et al. (2025); Chen et al. (2024b). Despite their progress, both paradigms have key limitations. Modular pipelines often accumulate errors across stages, while end-to-end models remain hard to interpret and largely mimic driving behaviors without reasoning, leading to robustness and safety issues Badue et al. (2021); Pan et al. (2024); Chen et al. (2024b).

In recent years, vision-language models (VLMs), built on large language models (LLMs), demonstrate strong capabilities in zero-shot transfer and language-based reasoning, highlighting their potential for autonomous driving Achiam et al. (2023); Wang et al. (2024b); Team (2025); Chen et al. (2024c). A number of recent studies have begun to adapt VLMs to autonomous driving, ranging from open-loop scene understanding to language-guided closed-loop driving systems, but they still face challenges in reliability and real-time performance Tian et al. (2024); Sima et al. (2025).

Knowledge distillation (KD) transfers knowledge from large models to small ones, enabling efficient deployment on resource-constrained edge devices Xu et al. (2024a); Fang et al. (2026). However, current approaches lack systematic studies on distillation signal selection in autonomous driving, leaving limited guidance on where to apply supervision Feng et al. (2024); Yang et al. (2025). Moreover, many works treat driving capabilities as parallel objectives, ignoring their sequential dependencies Tian et al. (2024); Shao et al. (2024). When multiple distillation objectives are jointly optimized, a common practice is to aggregate them with a fixed or heuristic weighted sum, which

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.

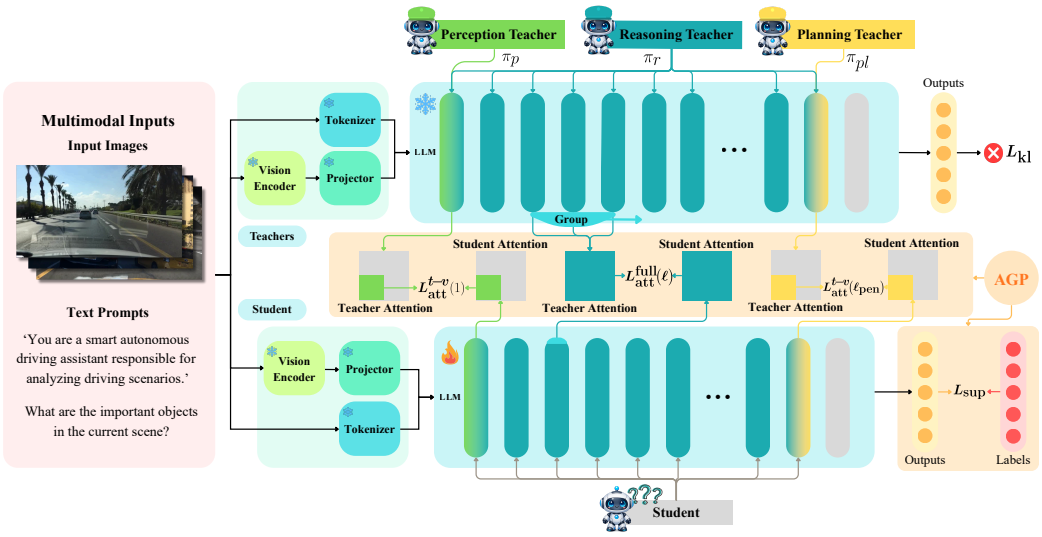


Figure 1: Drive-KD framework. Three capability-specific teachers distill layer-specific attention into one student, while AGP mitigates cross-capability gradient conflicts.

can improve one capability at the cost of degrading others Chen et al. (2024a); Zhong et al. (2024). Such trade-offs are unacceptable in autonomous driving, where perception, reasoning, and planning must be jointly reliable to ensure safe decision-making.

To address these limitations, we propose Drive-KD (Figure 1), a multi-teacher distillation framework for small-scale VLMs, aimed at improving their capability on comprehensive driving VQA tasks to better support high-level language guidance in autonomous driving. We decompose autonomous driving into a sequential triad of capabilities, "perception–reasoning–planning", where later stages build on earlier-stage information. Based on this capability decomposition, we first investigate layer selection strategies from both representation and capability perspectives. We then identify suitable supervision signals for capability transfer by comparing their dispersion, and analyze why output-distribution alignment is not suitable for autonomous driving.

Based on these findings, we systematically study distillation recipes for the three capabilities in autonomous driving, achieving capability scores that significantly outperform both pretrained and SFT baselines. These results further validate the layer and signal selection principles identified in our initial analyses. Moreover, we build Drive-KD as a multi-teacher distillation framework over the three capabilities, and apply asymmetric gradient projection (AGP) to mitigate gradient conflicts, yielding stronger capabilities in the small model; through attention analysis, we further find that Drive-KD restores the visual grounding suppressed by SFT. We further demonstrate the generality of Drive-KD across model scales and different model families.

We summarize our contributions as follows:

- First, we conduct the first systematic study of distillation design for autonomous driving VLMs, providing choices for distillation layers and supervision signals, and highlighting the limitations of output-distribution alignment for autonomous driving.
- Second, we derive the single-teacher distillation recipes for each capability and show stable improvements over pretrained and SFT baselines. We then develop Drive-KD as a multi-teacher distillation framework with AGP, and further verify its generality across model scales and different model families.
- Third, our distilled InternVL3-1B requires $\sim$42$\times$ less GPU memory and delivers $\sim$11.4$\times$ higher throughput while outperforming the same-family pretrained 78B model on DriveBench and surpassing GPT-5.1 on planning; on Bench2Drive-VL, it raises the closed-loop Driving Score from SFT's 37.36 to 62.41. Our attention analysis shows that Drive-KD restores capability-specific visual grounding suppressed by SFT.

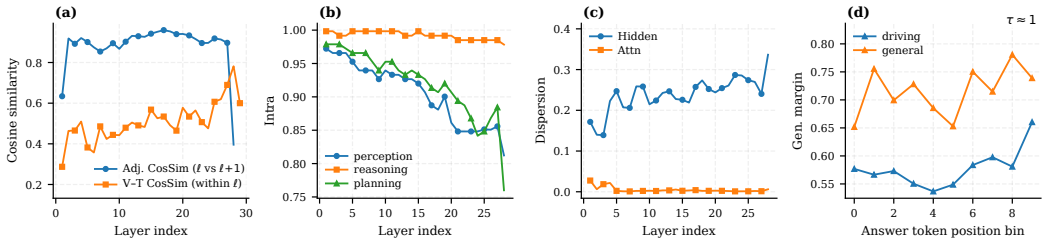


Figure 2: Pre-study summary for InternVL3-8B: (a) layer-wise cosine alignment (adjacent, vision–text), (b) intra-group similarity by capability, (c) dispersion of hidden states and attention $(1 - \cos)$, and (d) position-normalized margin on answer tokens for driving vs. general data at $\tau \approx 1.0$.

2 Related Work

2.1 LLM / VLM in Autonomous Driving

The rapid progress of LLMs has accelerated the development of multimodal models Zhao et al. (2023); Zhang et al. (2024). Building on this trend, VLMs have emerged and show strong general capabilities in complex scene description, visual question answering, and multi-step reasoning Lu et al. (2025); Wang et al. (2024b), which match the requirements of autonomous driving Zhou et al. (2024); You et al. (2026); Li et al. (2025a). Recent efforts explore VLM-based scene description and hierarchical planning, employ structured VQA for reasoning, and develop end-to-end driving systems with language-grounded low-level controls Tian et al. (2024); Sima et al. (2025); Shao et al. (2024); Xu et al. (2025); Hwang et al. (2025). However, current VLMs suffer from issues such as hallucination, unstable decision-making, and poor real-time performance, which limit their effectiveness in language-guided driving systems Guan et al. (2024); Li et al. (2025c).

2.2 Distillation for LLM / VLM

KD aims to transfer knowledge from a large teacher model to a smaller student model, improving the student's performance under limited capacity Xu et al. (2024a); Gou et al. (2021). It was originally dominated by output distribution alignment Hinton et al. (2015). In the era of LLMs, KD has gradually expanded to richer forms of supervision beyond outputs Li et al. (2023); Wang et al. (2024a). For VLMs, distillation needs to compress not only language modeling ability but also transfer cross-modal alignment and fusion Feng et al. (2024); Li et al. (2025b); Wen et al. (2025). Accordingly, recent works combine output-level alignment with internal-signal distillation and adopt techniques such as multi-teacher distillation to convey finer-grained information Li et al. (2023); Wang et al. (2024a). These VLM-oriented distillation methods have been widely applied to VQA compression and multimodal instruction following Tian et al. (2025); Cao et al. (2025). However, principled strategies for choosing distillation signals and coordinating distillation across tasks remain unclear Wang et al. (2024a); Mansourian et al. (2025).

3 Methodology

3.1 Preliminary

We categorize driving capabilities into three sequential capabilities: perception, reasoning, and planning. Perception aims to identify semantic cues that form the basis for subsequent reasoning. Reasoning leverages these cues to infer relations and constraints among entities in the environment. Planning then leverages the inferred information to determine safe driving actions. Section A provides supporting evidence for this capability organization.

3.2 Distillation Signal Selection

We investigate (i) the appropriate layers for distilling each capability and (ii) the specific signals to distill within the selected layers. In addition, we revisit output-distribution alignment and analyze

its limitations for autonomous driving. We sample 1,000 images, and each image contains multiple questions under each capability. Figure 2 summarizes these analyses and supports our algorithm design. Sections B.1 to B.4 provide the full pre-study results.

3.2.1 WHERE TO DISTILL: LAYER SELECTION

Distilling all layers is costly and unnecessary: layers encode different functions, and forcing the student to match the teacher everywhere can introduce redundant or conflicting supervision Liang et al. (2023). Instead, we select a set of informative layers for each capability. We select distillation layers from two complementary views: representation and capability. Combining representation change, cross-modal interaction, and intra-consistency, we identify stable capability layers.

Representation view. We use two layer-wise metrics: adjacent-layer cosine similarity of hidden states and within-layer vision–text cosine similarity. For Transformer layers $\ell = 1, \ldots, N$, let hidden states be $\mathbf{H}^{(\ell)} \in \mathbb{R}^{B\times L\times D}$, with token vectors $\mathbf{h}_{b,i}^{(\ell)} \in \mathbb{R}^D$ at position i of sample b. Treating the input embeddings as layer $\ell = 0$, we write $\mathbf{H}^{(0)} \in \mathbb{R}^{B\times L\times D}$ with token vectors $\mathbf{h}_{b,i}^{(0)}$, and the penultimate layer index is $\ell_{\text{pen}} = N - 1$. We define adjacent-layer similarity $S^{(\ell)}$ as the cosine similarity between hidden vectors at the same token position in layers ℓ and $\ell + 1$, averaged over samples and token indices. To measure intra-layer vision–text similarity, we split tokens into vision and text segments and mean-pool each segment for every sample. We then define $C^{(\ell)}$ as the sample-averaged cosine similarity between the pooled vision and text vectors. Across all models, $S^{(0)}$ is low while $S^{(1)}$ rises sharply, and $S^{(N-1)}$ drops relative to $S^{(N-2)}$, highlighting major representation changes near Layer 1 and ℓ_{pen}. Meanwhile, $C^{(\ell)}$ exhibits a clear fusion peak near the penultimate layer (ℓ_{pen}), indicating strong vision–text interaction. These results suggest Layer 1 and Layer ℓ_{pen} serve as key informative layers, and we treat them as primary candidates.

Capability view. The representation view is insufficient to identify layers that encode stable, capability-level structure across images, so we further use capability-wise intra-consistency as a complementary criterion for layer selection. On the sampled dataset, for each layer ℓ and capability c, we compute an image–capability center by averaging the hidden vectors of the last non-special token over all questions of capability c for the same image, $\mathbf{c}_{c,\text{img}_g}^{(\ell)} = N_{c,g}^{-1} \sum_{j=1}^{N_{c,g}} \mathbf{h}_{c,g,j}^{(\ell)}$. Here, $\mathbf{h}_{c,g,j}^{(\ell)}$ is the layer-ℓ hidden state of the last non-special token for the j-th question of capability c on image g. Let $\{\mathbf{c}_{c,\text{img}_1}^{(\ell)}, \ldots, \mathbf{c}_{c,\text{img}_G}^{(\ell)}\}$ be the centers of capability c across G images; we define intra as the mean pairwise cosine similarity, $\text{intra}_c^{(\ell)} = \frac{2}{G(G-1)} \sum_{i<j} \cos\Big(\mathbf{c}_{c,\text{img}_i}^{(\ell)}, \mathbf{c}_{c,\text{img}_j}^{(\ell)}\Big)$. Here, $\cos(\mathbf{a}, \mathbf{b}) = \langle \mathbf{a}/\|\mathbf{a}\|_2, \mathbf{b}/\|\mathbf{b}\|_2 \rangle$ and the summation is over all unordered image pairs ($i < j$).

Across all models, we observe consistent capability-specific patterns: perception has the highest intra at Layer 1 and then decreases with depth, reasoning keeps a high intra across layers, and planning decreases overall but shows a clear peak near Layer ℓ_{pen}; together with the representation view, these results support selecting Layer 1 for perception, using broader intermediate layers for reasoning to connect early perception cues with later decision features, and using the penultimate layer for planning to leverage late-stage capability specialization while retaining upstream information.

3.2.2 WHAT TO DISTILL: SIGNAL SELECTION

For each (image, question) instance, we extract (i) the last non-special-token hidden vector and (ii) the head-mean attention matrix at the target layer, and measure dispersion across questions for the same image using $1 - \cos(\cdot, \cdot)$, averaged over all question pairs and then over images. For hidden states, for any two questions i and j paired under the same image, we compute the dispersion as $D_{\text{hid}}(i, j) = 1 - \cos(u_i, u_j)$, where u_i (resp. u_j) is the hidden vector at the target layer for the last token of the generated answer segment excluding special tokens. For attention, we use the head-mean attention map $\bar{A}$ and define $D_{\text{att}}(i, j) = 1 - \cos(\text{vec}(\tilde{A}_i), \text{vec}(\tilde{A}_j))$, where $\tilde{A}_i$ and $\tilde{A}_j$ are the overlap-aligned top-left submatrices of $\bar{A}_i$ and $\bar{A}_j$ with size $m = \min(L_i, L_j)$, corresponding to the shared prefix tokens (vision tokens + system prompt), and are flattened by $\text{vec}(\cdot)$ before cosine.

Across all capabilities and models, attention shows lower dispersion than hidden states, indicating more stable capability-related behavior; thus, we use attention as the distillation signal.

3.2.3 REVISITING OUTPUT-DISTRIBUTION ALIGNMENT

Output-distribution alignment is implemented as KL divergence between the teacher and student predictive distributions. However, its effectiveness is highly sensitive to teacher capability and task complexity. Across multiple temperatures, we feed the teacher with both autonomous driving data and generic multimodal QA data Xu et al. (2024b), and quantify its confidence on answer tokens using top-1 probability, top-10 probability mass, and the generalized margin.

Let teacher logits at answer position t be $\mathbf{z}_t \in \mathbb{R}^{|\mathcal{V}|}$ and define the temperature-scaled softmax $p_t^{(\tau)}(k) = \mathrm{softmax}(\mathbf{z}_t/\tau)_k$; we compute top-1 confidence $m_t^{(\tau)} = \max_k p_t^{(\tau)}(k)$, top-10 mass $S_t^{(\tau)} = \sum_{i=1}^{10} p_t^{(\tau)}(k_i)$, and generalized margin $\Delta_{t,\mathrm{gen}}^{(\tau)} = m_t^{(\tau)} - \big(S_t^{(\tau)} - m_t^{(\tau)}\big)/(10-1)$. Here, $k_1, \ldots, k_{10}$ index the top-10 tokens under $p_t^{(\tau)}$. The generalized margin $\Delta_{t,\mathrm{gen}}^{(\tau)}$ measures the dominance of the top-1 token over other top-10 candidates; larger values indicate higher confidence. Across temperatures, driving outputs are less confident and more diffuse than general outputs, making output-distribution alignment a noisy distillation signal; thus, we exclude it from Drive-KD.

3.3 SINGLE-TEACHER DISTILLATION

3.3.1 DISTILLATION SIGNALS AND OBJECTIVES

We optimize the student with the hard-label supervised loss on our driving distillation dataset and add capability-specific attention distillation terms from a teacher.

Hard-label supervised loss. Given a token sequence $\{y_t\}_{t\in\mathcal{Y}_{\mathrm{ans}}}$ over the answer span $\mathcal{Y}_{\mathrm{ans}}$, the hard-label supervised loss is

$$L_{\mathrm{sup}} = -\frac{1}{|\mathcal{Y}_{\mathrm{ans}}|} \sum_{t\in\mathcal{Y}_{\mathrm{ans}}} \log p_\theta(y_t \mid x, y_{<t}). \tag{1}$$

Attention Maps. Let $A_{(\cdot)}^{(\ell)} \in \mathbb{R}^{B\times H_{(\cdot)}\times L\times L}$ denote the attention weights at layer ℓ, where $(\cdot) \in \{s,t\}$ indexes the student or teacher. We average over heads as $\bar{A}_{(\cdot),b,i,j}^{(\ell)} = H_{(\cdot)}^{-1} \sum_{h=1}^{H_{(\cdot)}} A_{(\cdot),b,h,i,j}^{(\ell)}$.

Full-attention distillation. We distill the full attention matrix by matching all query-to-key entries on valid positions between the student and teacher at layer ℓ, where i indexes the query token position and j indexes the key token position in a length-L sequence.

$$L_{\mathrm{att}}^{\mathrm{full}}(\ell) = \mathrm{mean}_{b,i}\left[\mathrm{mean}_j\left(\bar{A}_{s,b,i,j}^{(\ell)} - \bar{A}_{t,b,i,j}^{(\ell)}\right)^2\right]. \tag{2}$$

Compute the error for each query-token (b,i), average it over keys j, and then average over (b,i).

Text-to-vision attention distillation. We compute the loss over text query positions and normalize per sample by the number of text-to-vision pairs. Let T_b and V_b denote the sets of text-query and vision-key positions in sample b, respectively, where T_b includes all text tokens in the prompt+answer sequence excluding special and padding tokens. The distillation loss is

$$L_{\mathrm{att}}^{t\to v}(\ell) = \mathrm{mean}_b\ \frac{1}{|T_b|\,|V_b|} \sum_{i\in T_b}\sum_{j\in V_b}\left(\bar{A}_{s,b,i,j}^{(\ell)} - \bar{A}_{t,b,i,j}^{(\ell)}\right)^2. \tag{3}$$

We apply an online loss reweighting based on per-loss gradient magnitudes to stabilize multi-loss training. Further implementation details are provided in Section C.1. In this single-teacher stage, we do not use gradient projection, as it would dampen the attention-distillation gradients and thus confound a clean comparison among different distillation signals.

3.3.2 CAPABILITY-SPECIFIC RECIPES

We use hard-label supervision for all capabilities, and add capability-specific attention distillation as the soft term. For perception, we distill Layer 1 text-to-vision attention, $A_{1,t\to v}$. For reasoning, we distill intermediate-layer attention with layer-group matching Kim et al. (2025). Let the student layers be $\ell_1, \ldots, \ell_K$ and the teacher layers be $k_1, \ldots, k_M$, where both sets are restricted to the range from Layer 1 to the penultimate layer ℓ_{pen}; we set group size $g = \max(1,\ M - K + 1)$ and assign each student layer ℓ_i to a teacher group $\mathcal{G}(\ell_i) = \{k_{s_i+1}, \ldots, k_{s_i+g}\}$ with $s_i = \min(i-1,\ M-g)$, using mean attention over $\mathcal{G}(\ell_i)$ as the target. For planning, we distill penultimate-layer text-to-vision attention, $A_{\ell_{\text{pen}},t\to v}$.

3.4 MULTI-TEACHER DISTILLATION

3.4.1 OBJECTIVE AND TEACHER MIXING

Following the single-teacher recipes in Section 3.3.2, we define Drive-KD as a multi-teacher distillation framework with three teachers specialized for each capability, and we apply three teachers to every batch. For a batch from capability c, we combine the three teachers with a fixed weight $\pi_{c,t}$, which sets how much teacher t contributes on capability c (see Section C.2). Each teacher provides a capability-specific attention distillation loss L_t. We use the following definitions:

$$
\begin{aligned}
L_p = L_{\text{att}}^{t\to v}(1), \qquad L_{pl} = L_{\text{att}}^{t\to v}(\ell_{\text{pen}}), \quad \ell_{\text{pen}} = N_s - 1, \\
L_r = \frac{1}{|\mathcal{S}|} \sum_{\ell \in \mathcal{S}} \text{mean}_{b,i} \left[\text{mean}_j \left(\bar{A}_{s,b,i,j}^{(\ell)} - \mu_{t,b,i,j}^{(\ell)} \right)^2 \right], \\
\mu_{t,b,i,j}^{(\ell)} \ = \ \text{mean}_{k \in \mathcal{G}(\ell)}\, \bar{A}_{t,b,i,j}^{(k)}.
\end{aligned}
\tag{4}
$$

Here $\mathcal{S}$ is the selected student layer set, and $\mathcal{G}(\ell)$ maps each $\ell \in \mathcal{S}$ to a small group of nearby teacher layers. With these definitions, the total loss combines hard-label supervision with mixed distillation:

$$
L = w_{\text{ce}} L_{\text{sup}} + \sum_{t \in \{p,r,pl\}} \pi_{c,t}\, w_t\, L_t. \tag{5}
$$

3.4.2 ASYMMETRIC GRADIENT PROJECTION

To reduce gradient conflicts, we adopt AGP with two stages.

Stage 1: Asymmetric capability-wise projection and merging. For capability c, define $\mathbf{g}_{\text{sup}} = \nabla_\theta(w_{\text{ce}} L_{\text{sup}})$ and $\mathbf{g}_c = \nabla_\theta \Big(\sum_{t \in \{p,r,pl\}} \pi_{c,t}\, w_t\, L_t \Big)$; we use $\mathbf{g}_{\text{sup}}$ as anchor and $\mathbf{g}_c$ as follower. We set $\mathbf{g}_a^{(c)} = \mathbf{g}_{\text{sup}}$ and $\mathbf{g}_f = \mathbf{g}_c$. We first remove only the conflicting component of the follower along the anchor direction while keeping the anchor unchanged, and then merge them as $\mathbf{g}^{(c)} = \mathbf{g}_a^{(c)} + \mathbf{g}_f$:

$$
\mathbf{g}_f \leftarrow \mathbf{g}_f - \frac{\mathbf{g}_f^\top \mathbf{g}_a^{(c)}}{\|\mathbf{g}_a^{(c)}\|_2^2}\, \mathbf{g}_a^{(c)}, \quad \text{if } \mathbf{g}_f^\top \mathbf{g}_a^{(c)} < 0 \tag{6}
$$

Stage 2: Symmetric pairwise projections among merged capability gradients. We have three capabilities $\mathcal{C} = \{p, r, pl\}$ and obtain merged gradients $\mathbf{g}^{(p)}, \mathbf{g}^{(r)}, \mathbf{g}^{(pl)}$ from Stage 1. To mitigate cross-capability conflicts, we initialize $\tilde{\mathbf{g}}^{(c_i)} \leftarrow \mathbf{g}^{(c_i)}$ for $i \in \{1,2,3\}$, check conflicts in a shuffled pairwise order, and apply a projection only when a conflict is detected. Specifically, we sample a random permutation (c_1, c_2, c_3) of (p, r, pl) and sequentially update each $\tilde{\mathbf{g}}^{(c_i)}$ by projecting it against the other two gradients in the order $c_1 \to c_2 \to c_3$.

$$
\tilde{\mathbf{g}}^{(c_i)} \leftarrow \tilde{\mathbf{g}}^{(c_i)} - \frac{\tilde{\mathbf{g}}^{(c_i)\top} \mathbf{g}^{(c_j)}}{\|\mathbf{g}^{(c_j)}\|_2^2}\, \mathbf{g}^{(c_j)}, \quad \text{if } \tilde{\mathbf{g}}^{(c_i)\top} \mathbf{g}^{(c_j)} < 0 \tag{7}
$$

Finally, we sum the projected capability gradients to form the overall update $\mathbf{g} = \tilde{\mathbf{g}}^{(p)} + \tilde{\mathbf{g}}^{(r)} + \tilde{\mathbf{g}}^{(pl)}$.

Stage 1 uses the hard-label supervised gradient as an anchor and removes the part of the distillation update that conflicts with it, avoiding sacrificing hard-label learning to fit weaker teachers' noisier

Table 1: DriveBench capability and deployment efficiency of representative VLMs.

Model	Capability scores (%)				Deployment metrics		
	Perception	Reasoning	Planning	Avg.	Memory (GB)	Speed (tok/s)	1st-token (s)
GPT-5.1	45.56	41.02	51.94	**46.17**	-	-	-
InternVL3-1B	33.26	20.96	22.36	25.53	**4.1**	**45.7**	**0.45**
InternVL3-2B	37.71	35.99	26.19	33.30	6.3	39.9	0.67
InternVL3-8B	40.05	41.15	32.77	37.99	18.3	32.6	1.58
InternVL3-14B	39.83	40.84	36.35	39.01	33.4	17.0	3.01
InternVL3-38B	34.27	38.48	40.65	37.80	87.0	7.3	9.84
InternVL3-78B	42.01	**47.16**	36.31	41.83	171.6	4.0	16.46
Qwen2.5-VL-3B-Instruct	35.46	30.81	25.29	30.52	8.5	28.0	0.68
Qwen2.5-VL-3B-DriveLM	30.19	35.28	35.41	33.63	8.5	28.0	0.68
Qwen2.5-VL-7B-Instruct	36.26	37.54	32.84	35.55	17.1	32.0	0.87
Qwen2.5-VL-32B-Instruct	38.41	41.30	34.29	38.00	69.5	10.8	2.36
Qwen2.5-VL-72B-Instruct	23.78	27.67	50.76	34.07	146.5	5.8	4.26
Llama-3.2-11B-Vision-Instruct	31.59	32.91	29.34	31.28	26.2	16.8	1.55
Llama-3.2-90B-Vision-Instruct	27.26	26.33	27.72	27.10	183.6	2.7	8.05
InternVL3-1B (SFT)	39.61	25.74	45.48	36.94	**4.1**	**45.7**	**0.45**
Qwen2.5-VL-3B-Instruct (SFT)	42.91	33.94	47.13	41.33	8.5	28.0	0.68
InternVL3-1B (Single)	43.13	34.32	52.97	43.47	**4.1**	**45.7**	**0.45**
Qwen2.5-VL-3B-Instruct (Single)	45.59	34.47	51.97	44.01	8.5	28.0	0.68
InternVL3-1B (Multi)	43.50	33.15	**55.51**	44.05	**4.1**	**45.7**	**0.45**
Qwen2.5-VL-3B-Instruct (Multi)	**45.63**	36.41	54.07	45.37	8.5	28.0	0.68

or mismatched soft signals. Stage 2 resolves negative inner products among capability-level merged gradients to reduce the chance of improving one capability at the expense of another; the shuffled projection order helps avoid consistently biasing toward any fixed capability. An intuitive illustration and empirical analysis of how AGP resolves gradient conflicts are provided in Section D.

4 Experiments

4.1 Experimental Setup

Data and prompts. We train on a 10,500 human-annotated driving distillation dataset built from nuScenes and BDD100K images Caesar et al. (2020); Yu et al. (2020a), with single- and multi-view VQA; Section E.2 reports diversity statistics and Section E.3 provides examples.

Models and training. Our main experiments consider two teacher–student pairs: InternVL3-8B $\rightarrow$ InternVL3-1B and Qwen2.5-VL-7B-Instruct $\rightarrow$ Qwen2.5-VL-3B-Instruct (see Section F.1). Detailed training configurations are provided in Section F.

Evaluation. We conduct open-loop evaluation on DriveBench Xie et al. (2025) and Drive-P2D Tang et al. (2026), and closed-loop evaluation on Bench2Drive-VL Jia et al. (2026). Evaluation settings (including the DeepSeek-V3.2 judge DeepSeek-AI et al. (2025)), Drive-P2D results, and comparisons with representative KD methods are reported in Sections G.1 to G.3, respectively.

4.2 Single-Teacher Evaluation

We evaluate single-teacher distillation by distilling each student from a single teacher within its model family. Each capability-specific single-teacher student is trained only on the corresponding capability split of our distillation dataset (Section E.2).

As shown in Table 1, single-teacher distillation raises Avg. from 25.53 to 43.47 for InternVL3-1B and from 30.52 to 44.01 for Qwen2.5-VL-3B-Instruct. The two students score 52.97 and 51.97 on

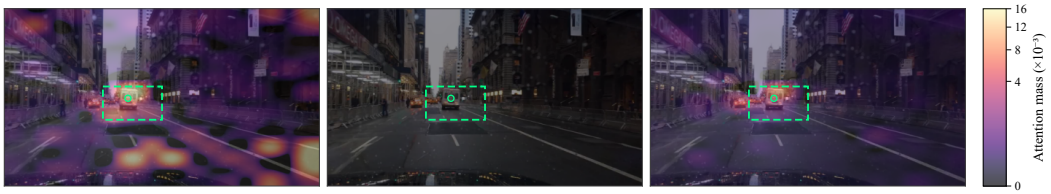


Figure 3: Visual grounding with Drive-KD. On a driving QA example, Drive-KD attends more clearly to task-relevant visual evidence than Base and SFT.

Table 2: Bench2Drive-VL closed-loop results. DS/RC denote Driving Score/Route Completion.

Model	DS ↑	RC ↑
InternVL3-1B	2.81	3.00
InternVL3-1B (SFT)	37.36	71.52
InternVL3-1B (Multi)	62.41	84.50
Qwen2.5-VL-3B-Instruct	24.91	42.78
Qwen2.5-VL-3B-Instruct (SFT)	49.68	85.24
Qwen2.5-VL-3B-Instruct (Multi)	**69.76**	**92.84**

Table 3: DriveBench scores (%) for InternVL3 teacher–student size pairs.

Tea	Stu	Perception	Reasoning	Planning	Avg.
8B	1B	**43.50**	33.15	55.51	44.05
14B	1B	43.41	30.34	56.19	43.31
38B	1B	43.24	29.15	56.77	43.05
8B	2B	43.14	36.97	56.01	45.37
14B	2B	41.74	35.40	56.84	44.66
38B	2B	42.87	**38.25**	**57.63**	**46.25**

planning, slightly above GPT-5.1 at 51.94 while outperforming ReasonDrive Chahe & Zhou (2025). This shows capability-specific attention distillation effectively transfers driving behaviors.

By capability, perception improves strongly, supporting early cross-modal attention supervision. Reasoning improves but remains the hardest to transfer, reflecting the difficulty of multi-step relational understanding. Planning gains the most and becomes the students' strongest capability, supporting late-stage cross-modal attention for decision supervision.

4.3 Multi-Teacher Evaluation

Compared with the single-teacher, multi-teacher training achieves a higher overall score (see Table 1). For InternVL3-1B, it improves Avg. from 43.47 to 44.05 and planning from 52.97 to 55.51, while preserving strong perception, highlighting the complementary benefits of Drive-KD.

By capability, perception stays on par with single-teacher, suggesting early-layer cross-modal supervision is sufficient. Reasoning remains comparable but appears most sensitive to cross-capability interference under multi-objective training. Planning improves the most and drives the higher average score, consistent with its reliance on integrating perception cues with intermediate reasoning signals, which multi-teacher supervision strengthens. Although our students do not consistently surpass GPT-5.1 in perception and reasoning, they exhibit substantially stronger reasoning–planning alignment, indicating a more coherent transition from scene understanding to driving decisions (Section H.1). In terms of efficiency, the distilled models retain small-VLM deployment efficiency while achieving stronger capability scores. For example, InternVL3-1B (Multi) uses 4.1 GB peak memory at 45.7 tok/s, yielding a better capability–efficiency trade-off than InternVL3-78B.

Our attention analysis further shows that Drive-KD restores the visual grounding suppressed by SFT across all three capabilities: it sharpens perception around decisive cues, while enabling reasoning and planning to integrate broader relational and scene-level evidence. A representative example is shown in Figure 3, with an additional example and full quantitative results in Section H.2.

4.4 Closed-Loop Driving Evaluation

Unlike open-loop evaluation of isolated answers, closed-loop driving feeds decisions back into subsequent states, testing whether capability gains translate into executable behavior under interaction and accumulating errors. Compared with SFT, Drive-KD improves Driving Score by 25.05 and 20.08 and Route Completion by 12.98 and 7.60 across both families, showing consistent gains.

Table 4: Partial ablation results.

Setting	P	R	Pl
CE (SFT)	40.86	29.05	45.63
CE + KL	39.60	28.16	43.36
CE + Hidden (1)	41.27	-	-
CE + Hidden (mid)	-	31.65	-
CE + Hidden (ℓ_{pen})	-	-	45.04
Ours (single-teacher)	**43.13**	**34.32**	**52.97**
Multi-teacher	42.34	25.68	51.03
Multi-teacher (G1)	42.96	25.49	46.99
Multi-teacher (G2)	42.64	29.18	52.19
Ours (multi-teacher)	**43.50**	**33.15**	**55.51**

In closed-loop driving, errors in different capabilities can propagate along the capability chain, while actions alter subsequent observations and feed errors into decisions. Hard-label SFT poorly constrains this process. Drive-KD instead transfers internal signals for three capabilities across layers and strengthens them through multi-teacher training, enabling coordination under feedback and converting transferred visual-language capabilities into more reliable route execution.

4.5 Distillation Scaling

We run the multi-teacher algorithm with teacher sizes $\{8\text{B}, 14\text{B}, 38\text{B}\}$ and student sizes $\{1\text{B}, 2\text{B}\}$. Table 3 shows that for a fixed student, larger teachers can improve, match, or degrade different capabilities, and the best teacher size depends on student capacity. Under comparable distillation settings, the higher-capacity student is consistently stronger, indicating that student capacity remains crucial for absorbing knowledge, and stronger teachers may be harder for small students to learn due to more complex feature representations. Drive-KD remains effective across different size pairings.

4.6 Ablation Study

We ablate six aspects on InternVL3-1B: (i) whether to add output-distribution alignment (CE+KL) beyond hard-label CE, (ii) hidden-state distillation variants, and (iii) gradient-conflict handling in multi-teacher training (none, G1/G2, or AGP). Here (i) and (ii) follow the single-teacher protocol: the model is trained on the capability split only, and we report the score on that capability (Table 4). Section I reports the remaining ablations in more detail, including per-capability layer choices, alternative attention targets and similarity-based attention matching.

CE is equivalent to SFT, and KL aligns the output token distribution. "mid" denotes the intermediate layers from the first to the penultimate; G1 applies PCGrad Yu et al. (2020b) jointly across all losses, while G2 projects CE vs. soft within each capability and then projects across capabilities.

Overall, KL consistently degrades performance relative to CE, and comparisons against hidden-state distillation further support our signal choices; in contrast, our single-teacher recipe achieves the best overall performance. In multi-teacher training, the plain baseline severely hurts reasoning, and joint PCGrad (G1) is also unstable—again most clearly on reasoning. A two-stage symmetric scheme (G2) recovers reasoning and improves planning, but AGP achieves the best planning score while keeping perception and reasoning strong. Together, these ablations show that our capability-specific distillation signals and AGP are both critical for consistent gains.

5 Conclusion

We propose Drive-KD, a multi-teacher distillation framework for driving VLMs. Drive-KD decomposes driving into a sequential triad of perception, reasoning, and planning, and transfers each capability with targeted internal supervision. First, we study distillation design for driving VLMs. We identify Layer 1 cross-modal attention as most informative for perception, broader intermediate layers for reasoning, and the penultimate layer for planning. We find attention is more stable than hidden states, and output-distribution alignment is unreliable for autonomous driving. Second, based on these findings, we derive strong single-teacher recipes for each capability and improve small VLMs over baselines. Third, we unify the three capabilities into a multi-teacher framework and introduce AGP to reduce cross-capability interference. Finally, experiments show distilled models achieve strong capability–efficiency trade-offs and Drive-KD generalizes across model scales and families. Our distilled InternVL3-1B uses $\sim$42$\times$ less peak GPU memory and achieves $\sim$11.4$\times$ higher generation throughput than pretrained InternVL3-78B, while outperforming it on DriveBench overall and surpassing GPT-5.1 on planning; on Bench2Drive-VL, Drive-KD raises the closed-loop Driving Score from SFT's 37.36 to 62.41 and achieves 84.50 Route Completion. Attention analysis shows that Drive-KD restores the capability-specific visual grounding suppressed by SFT. We hope Drive-KD enables efficient driving VLMs and better supports language-guided driving systems.

AI USE STATEMENT

Generative AI tools were used to assist with translation and to edit the research paper to improve readability. The tools were not used to generate synthetic data sets; help develop theoretical models or conceptual frameworks; formulate mathematical claims; provide critical ingredients for proving mathematical claims; assist in the writing of proofs; propose or refine hypotheses; design or provide feedback on research methodology or experiments; implement methods; clean or reformat datasets; support qualitative or thematic data analysis; interpret results; formulate questions for surveys or interviews; create or modify scientific figures or images; suggest experimental parameters; create or edit software code; create artifacts; draft parts of the research paper; transcribe recordings of research material; summarize or analyse existing literature; discover research topics or identify research gaps; brainstorm; source or search for information; identify relevant literature; format references; suggest a structure for the research paper; or propose a title or keywords. All AI-assisted text was reviewed by the authors, who take full responsibility for the final content of this submission.

ETHICS STATEMENT

Drive-KD is expected to achieve faster on-vehicle response while improving driving capability. By lowering the hardware requirements for advanced AI safety functions, it may help make such functions available across a wider range of vehicle classes and price points. Compared with traditional end-to-end "black-box" models, our "perception–reasoning–planning" triad offers better interpretability. Many recent language-grounded driving and vision-language-action (VLA) systems are fundamentally built upon vision-language models (VLMs). A significant portion of the challenges in closed-loop driving arises not only from control, but from limitations in upstream visual understanding, relational reasoning, and semantic decision-making. By improving the efficiency and capability of driving-oriented VLMs through structured distillation, our approach targets this critical upstream bottleneck and serves as a practical step toward more reliable closed-loop systems. However, reasoning ability remains the most difficult aspect to transfer to smaller models. Although we evaluate Drive-KD in both open-loop and closed-loop settings, these results do not guarantee safety in real-world deployment. We therefore recommend broader long-tail, edge-case, and physical-system validation before safety-critical deployment.

REPRODUCIBILITY STATEMENT

To facilitate reproducibility, Section 3 specifies the capability-specific distillation objectives and asymmetric gradient projection; Sections E.2 and E.3 document the dataset and representative annotations; Section F.1 reports the training setup and hyperparameters; and Sections G.1 and G.2 describe the evaluation protocols and additional benchmark results.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Claudine Badue, Rânik Guidolini, Raphael Vivacqua Carneiro, Pedro Azevedo, Vinicius B Cardoso, Avelino Forechi, Luan Jesus, Rodrigo Berriel, Thiago M Paixao, Filipe Mutz, et al. Self-driving cars: A survey. *Expert systems with applications*, 165:113816, 2021.

Mariusz Bojarski, Davide Del Testa, Daniel Dworakowski, Bernhard Firner, Beat Flepp, Prasoon Goyal, Lawrence D Jackel, Mathew Monfort, Urs Muller, Jiakai Zhang, et al. End to end learning for self-driving cars. *arXiv preprint arXiv:1604.07316*, 2016.

Holger Caesar, Varun Bankiti, Alex H. Lang, Sourabh Vora, Venice Erin Liong, Qiang Xu, Anush Krishnan, Yu Pan, Giancarlo Baldan, and Oscar Beijbom. nuscenes: A multimodal dataset for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020.

Jiajun Cao, Yuan Zhang, Tao Huang, Ming Lu, Qizhe Zhang, Ruichuan An, Ningning Ma, and Shanghang Zhang. MoVE-KD: Knowledge Distillation for VLMs with Mixture of Visual En-

coders. *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 19846–19856, 2025. URL https://api.semanticscholar.org/CorpusID:275324083.

Amirhosein Chahe and Lifeng Zhou. Reasondrive: Efficient visual question answering for autonomous vehicles with reasoning-enhanced small vision-language models, 2025. URL https://arxiv.org/abs/2504.10757.

Hongzhan Chen, Xiaojun Quan, Hehong Chen, Ming Yan, and Ji Zhang. Knowledge distillation for closed-source language models, 2024a. URL https://openreview.net/forum?id=oj4KUNoKnf.

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

Zhe Chen, Jiannan Wu, Wenhai Wang, Weijie Su, Guo Chen, Sen Xing, Muyan Zhong, Qinglong Zhang, Xizhou Zhu, Lewei Lu, et al. Internvl: Scaling up vision foundation models and aligning for generic visual-linguistic tasks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 24185–24198, 2024c.

DeepSeek-AI, Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, et al. DeepSeek-V3.2: Pushing the frontier of open large language models. *arXiv preprint arXiv:2512.02556*, 2025.

Luyang Fang, Xiaowei Yu, Jiazhang Cai, Yongkai Chen, Shushan Wu, Zhengliang Liu, Zhenyuan Yang, Haoran Lu, Xilin Gong, Yufang Liu, Terry Ma, Wei Ruan, Ali Abbasi, Jing Zhang, Tao Wang, Ehsan Latif, Weihang You, Hanqi Jiang, Wei Liu, Wei Zhang, Soheil Kolouri, Xiaoming Zhai, Dajiang Zhu, Wenxuan Zhong, Tianming Liu, and Ping Ma. Knowledge distillation and dataset distillation of large language models: Emerging trends, challenges, and future directions, 2026. URL https://arxiv.org/abs/2504.14772.

Qianhan Feng, Wenshuo Li, Tong Lin, and Xinghao Chen. Align-KD: Distilling Cross-Modal Alignment Knowledge for Mobile Vision-Language Model, December 2024. URL http://arxiv.org/abs/2412.01282. arXiv:2412.01282 [cs].

Sonda Fourati, Wael Jaafar, Noura Baccar, Safwan Alfattani, and Rami Langar. Foundation models for autonomous driving: A comprehensive survey. *Engineering Applications of Artificial Intelligence*, 176:114805, 2026. ISSN 0952-1976. doi: https://doi.org/10.1016/j.engappai.2026.114805. URL https://www.sciencedirect.com/science/article/pii/S0952197626010870.

Jianping Gou, Baosheng Yu, Stephen J Maybank, and Dacheng Tao. Knowledge distillation: A survey. *International journal of computer vision*, 129(6):1789–1819, 2021.

Tianrui Guan, Fuxiao Liu, Xiyang Wu, Ruiqi Xian, Zongxia Li, Xiaoyu Liu, Xijun Wang, Lichang Chen, Furong Huang, Yaser Yacoob, et al. Hallusionbench: an advanced diagnostic suite for entangled language hallucination and visual illusion in large vision-language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 14375–14385, 2024.

Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

Jun Hu, Yuefeng Wang, Shuai Cheng, Jinghan Xu, Ningjia Wang, Bingjie Fu, Zuotao Ning, Jingyao Li, Hualin Chen, Chaolu Feng, and Yin Zhang. A survey of decision-making and planning methods for self-driving vehicles. *Frontiers in Neurorobotics*, Volume 19 - 2025, 2025. ISSN 1662-5218. doi: 10.3389/fnbot.2025.1451923. URL https://www.frontiersin.org/journals/neurorobotics/articles/10.3389/fnbot.2025.1451923.

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

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, September 2025. URL http://arxiv.org/abs/2410.23262. arXiv:2410.23262 [cs].

Xiaosong Jia, Yuqian Shao, Zhenjie Yang, Qifeng Li, Zhiyuan Zhang, and Junchi Yan. Bench2drive-vl: Benchmarks for closed-loop autonomous driving with vision-language models, 2026. URL https://arxiv.org/abs/2604.01259.

Sicong Jiang, Zilin Huang, Kangan Qian, Ziang Luo, Tianze Zhu, Yang Zhong, Yihong Tang, Menglin Kong, Yunlong Wang, Siwen Jiao, Hao Ye, Zihao Sheng, Xin Zhao, Tuopu Wen, Zheng Fu, Sikai Chen, Kun Jiang, Diange Yang, Seongjin Choi, and Lijun Sun. A survey on vision-language-action models for autonomous driving, 2025. URL https://arxiv.org/abs/2506.24044.

Jiwan Kim, Kibum Kim, Sangwoo Seo, and Chanyoung Park. CompoDistill: Attention Distillation for Compositional Reasoning in Multimodal LLMs, October 2025. URL http://arxiv.org/abs/2510.12184. arXiv:2510.12184 [cs].

Peizheng Li, Zhenghao Zhang, David Holtz, Hang Yu, Yutong Yang, Yuzhi Lai, Rui Song, Andreas Geiger, and Andreas Zell. Spacedrive: Infusing spatial awareness into vlm-based autonomous driving. *arXiv preprint arXiv:2512.10719*, 2, 2025a.

Xuanlin Li, Yunhao Fang, Minghua Liu, Zhan Ling, Zhuowen Tu, and Hao Su. Distilling large vision-language model with out-of-distribution generalizability. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pp. 2492–2503, 2023.

Yuqi Li, Chuanguang Yang, Junhao Dong, Zhengtao Yao, Haoyan Xu, Zeyu Dong, Hansheng Zeng, Zhulin An, and Yingli Tian. AMMKD: Adaptive Multimodal Multi-teacher Distillation for Lightweight Vision-Language Models, August 2025b. URL http://arxiv.org/abs/2509.00039. arXiv:2509.00039 [cs].

Zongxia Li, Xiyang Wu, Hongyang Du, Huy Nghiem, and Guangyao Shi. Benchmark evaluations, applications, and challenges of large vision language models: A survey. *arXiv preprint arXiv:2501.02189*, 1, 2025c.

Chen Liang, Simiao Zuo, Qingru Zhang, Pengcheng He, Weizhu Chen, and Tuo Zhao. Less is more: Task-aware layer-wise distillation for language model compression. In *International Conference on Machine Learning*, pp. 20852–20867. PMLR, 2023.

Dongchen Lu, Yuyao Sun, Zilu Zhang, Leping Huang, Jianliang Zeng, Mao Shu, and Huo Cao. Internvl-x: Advancing and accelerating internvl series with efficient visual token compression. *arXiv preprint arXiv:2503.21307*, 2025.

Amir M Mansourian, Rozhan Ahmadi, Masoud Ghafouri, Amir Mohammad Babaei, Elaheh Badali Golezani, Zeynab Yasamani Ghamchi, Vida Ramezanian, Alireza Taherian, Kimia Dinashi, Amirali Miri, et al. A comprehensive survey on knowledge distillation. *arXiv preprint arXiv:2503.12067*, 2025.

Fenglian Pan, Yinwei Zhang, Jian Liu, Larry Head, Maria Elli, and Ignacio Alvarez. Reliability modeling for perception systems in autonomous vehicles: A recursive event-triggering point process approach. *Transportation Research Part C: Emerging Technologies*, 169:104868, 2024.

Hao Shao, Yuxuan Hu, Letian Wang, Guanglu Song, Steven L. Waslander, Yu Liu, and Hongsheng Li. LMDrive: Closed-Loop End-to-End Driving with Large Language Models. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 15120–15130, Seattle, WA, USA, June 2024. IEEE. ISBN 979-8-3503-5300-6. doi: 10.1109/CVPR52733.2024.01432. URL https://ieeexplore.ieee.org/document/10657019/.

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, January 2025. URL http://arxiv.org/abs/2312.14150. arXiv:2312.14150 [cs].

Jingchen Sun, Shaobo Han, Deep Patel, Wataru Kohno, Can Jin, and Changyou Chen. Uncertainty-aware knowledge distillation for multimodal large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Zecong Tang, Zixu Wang, Yifei Wang, Weitong Lian, Tianjian Gao, Haoran Li, Tengju Ru, Lingyi Meng, Zhejun Cui, Yichen Zhu, Qi Kang, Kaixuan Wang, and Yu Zhang. Drive-p2d: A progressive perception-to-decision benchmark for vlms in autonomous driving, 2026. URL https://arxiv.org/abs/2601.14702.

Qwen Team. Qwen2.5-vl, January 2025. URL https://qwenlm.github.io/blog/qwen2.5-vl/.

Huiyuan Tian, Bonan Xu, Shijian Li, and Xin Jin. From Low-Rank Features to Encoding Mismatch: Rethinking Feature Distillation in Vision Transformers, November 2025. URL http://arxiv.org/abs/2511.15572. arXiv:2511.15572 [cs].

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, June 2024. URL http://arxiv.org/abs/2402.12289. arXiv:2402.12289 [cs].

Jin Wang, Dawei Liao, You Zhang, Dan Xu, and Xuejie Zhang. Layerwised multimodal knowledge distillation for vision-language pretrained model. *Neural Networks*, 175:106272, 2024a.

Peng Wang, Shuai Bai, Sinan Tan, Shijie Wang, Zhihao Fan, Jinze Bai, Keqin Chen, Xuejing Liu, Jialin Wang, Wenbin Ge, et al. Qwen2-vl: Enhancing vision-language model's perception of the world at any resolution. *arXiv preprint arXiv:2409.12191*, 2024b.

Zichen Wen, Shaobo Wang, Yufa Zhou, Junyuan Zhang, Qintong Zhang, Yifeng Gao, Zhaorun Chen, Bin Wang, Weijia Li, Conghui He, and Linfeng Zhang. Efficient multi-modal large language models via progressive consistency distillation, 2025. URL https://arxiv.org/abs/2510.00515.

Shaoyuan Xie, Lingdong Kong, Yuhao Dong, Chonghao Sima, Wenwei Zhang, Qi Alfred Chen, Ziwei Liu, and Liang Pan. Are vlms ready for autonomous driving? an empirical study from the reliability, data and metric perspectives. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6585–6597, October 2025.

Xiaohan Xu, Ming Li, Chongyang Tao, Tao Shen, Reynold Cheng, Jinyang Li, Can Xu, Dacheng Tao, and Tianyi Zhou. A Survey on Knowledge Distillation of Large Language Models, October 2024a. URL http://arxiv.org/abs/2402.13116. arXiv:2402.13116 [cs].

Zhenhua Xu, Yan Bai, Yujia Zhang, Zhuoling Li, Fei Xia, Kwan-Yee K Wong, Jianqiang Wang, and Hengshuang Zhao. Drivegpt4-v2: Harnessing large language model capabilities for enhanced closed-loop autonomous driving. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, pp. 17261–17270, 2025.

Zhiyang Xu, Chao Feng, Rulin Shao, Trevor Ashby, Ying Shen, Di Jin, Yu Cheng, Qifan Wang, and Lifu Huang. Vision-flan: Scaling human-labeled tasks in visual instruction tuning, 2024b. URL https://arxiv.org/abs/2402.11690.

Xiaoyu Yang, Jie Lu, and En Yu. Learning from All: Concept Alignment for Autonomous Distillation from Multiple Drifting MLLMs, October 2025. URL http://arxiv.org/abs/2510.04142. arXiv:2510.04142 [cs].

Junwei You, Zhuoyu Jiang, Zilin Huang, Haotian Shi, Rui Gan, Keshu Wu, Xi Cheng, Xiaopeng Li, and Bin Ran. V2x-vlm: End-to-end v2x cooperative autonomous driving through large vision-language models. *Transportation Research Part C: Emerging Technologies*, 183:105457, 2026.

Fisher Yu, Haofeng Chen, Xin Wang, Wenqi Xian, Yingying Chen, Fangchen Liu, Vashisht Madhavan, and Trevor Darrell. Bdd100k: A diverse driving dataset for heterogeneous multitask learning. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020a.

Tianhe Yu, Saurabh Kumar, Abhishek Gupta, Sergey Levine, Karol Hausman, and Chelsea Finn. Gradient surgery for multi-task learning. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin (eds.), *Advances in Neural Information Processing Systems*, volume 33, pp. 5824–5836. Curran Associates, Inc., 2020b. URL https://proceedings.neurips.cc/paper_files/paper/2020/file/3fe78a8acf5fda99de95303940a2420c-Paper.pdf.

Jingyi Zhang, Jiaxing Huang, Sheng Jin, and Shijian Lu. Vision-language models for vision tasks: A survey. *IEEE transactions on pattern analysis and machine intelligence*, 46(8):5625–5644, 2024.

Jingyuan Zhao, Wenyi Zhao, Bo Deng, Zhenghong Wang, Feng Zhang, Wenxiang Zheng, Wanke Cao, Jinrui Nan, Yubo Lian, and Andrew F. Burke. Autonomous driving system: A comprehensive survey. *Expert Systems with Applications*, 242:122836, 2024. ISSN 0957-4174. doi: https://doi.org/10.1016/j.eswa.2023.122836. URL https://www.sciencedirect.com/science/article/pii/S0957417423033389.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, et al. A survey of large language models. *arXiv preprint arXiv:2303.18223*, 1(2), 2023.

Qihuang Zhong, Liang Ding, Li Shen, Juhua Liu, Bo Du, and Dacheng Tao. Revisiting Knowledge Distillation for Autoregressive Language Models, June 2024. URL http://arxiv.org/abs/2402.11890. arXiv:2402.11890 [cs].

Xingcheng Zhou, Mingyu Liu, Ekim Yurtsever, Bare Luka Zagar, Walter Zimmer, Hu Cao, and Alois C Knoll. Vision language models in autonomous driving: A survey and outlook. *IEEE Transactions on Intelligent Vehicles*, 2024.

A Capability Decomposition Evidence

We organize driving capabilities into perception, reasoning, and planning, following the sequential dependency of driving decisions. Since planning ultimately determines driving actions, we examine its alignment with preceding capabilities, namely perception and reasoning. To examine this alignment, we train four InternVL3-1B SFT variants: planning only, perception+planning, reasoning+planning, and perception+reasoning+planning. On 200 images, we aggregate image-level scores and compute Pearson and Spearman correlations between perception and planning, and between reasoning and planning. Table 5 reports the image-level correlations under these capability-composition settings.

Table 5: Image-level correlations under different capability-composition SFT settings.

Setting	Pear. (P, Pl)	Spear. (P, Pl)	Pear. (R, Pl)	Spear. (R, Pl)
Plan	+0.1599	+0.1492	-0.0498	-0.0160
Perc.+Plan	+0.2234	+0.2243	–	–
Reas.+Plan	–	–	+0.1144	+0.0923
Perc.+Reas.+Plan	**+0.2516**	**+0.2359**	**+0.1712**	**+0.1605**

The correlations become stronger as preceding capabilities are included during training, suggesting that planning is more aligned with perception and reasoning when those supporting capabilities are learned jointly. This provides empirical evidence supporting the capability decomposition used in Drive-KD.

B Additional Pre-study Analyses

B.1 Layer-wise similarity profiles across model families and scales

We report additional layer-wise similarity profiles (Figure 4) to support our layer selection criteria across model families and parameter scales.

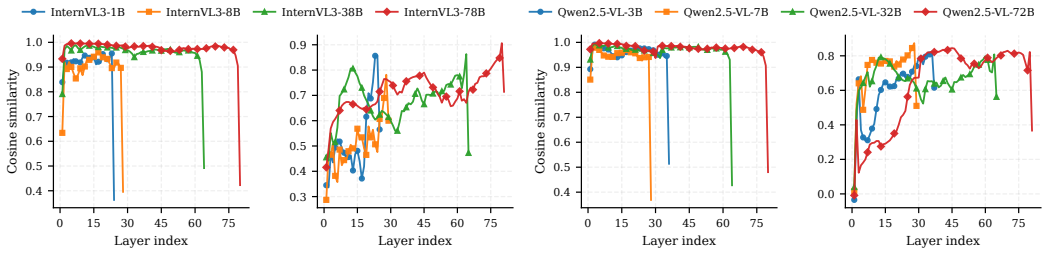


Figure 4: Layer-wise similarity profiles across model families and scales. Adjacent-layer cosine similarity (Adj. CosSim) and within-layer vision–text cosine similarity (V–T CosSim) are shown for InternVL3 (1B/8B/38B/78B) and Qwen2.5-VL (3B/7B/32B/72B).

B.2 Capability-wise intra-consistency across Transformer layers

We further include capability-wise intra-consistency curves (Figure 5), which complement representation-based signals by quantifying how stable each capability representation is across images at different depths.

B.3 Hidden-state vs. attention dispersion under rewording

We provide dispersion comparisons between hidden states and attention maps under question rewording (Figure 6) to justify using attention as a more capability-stable distillation signal.

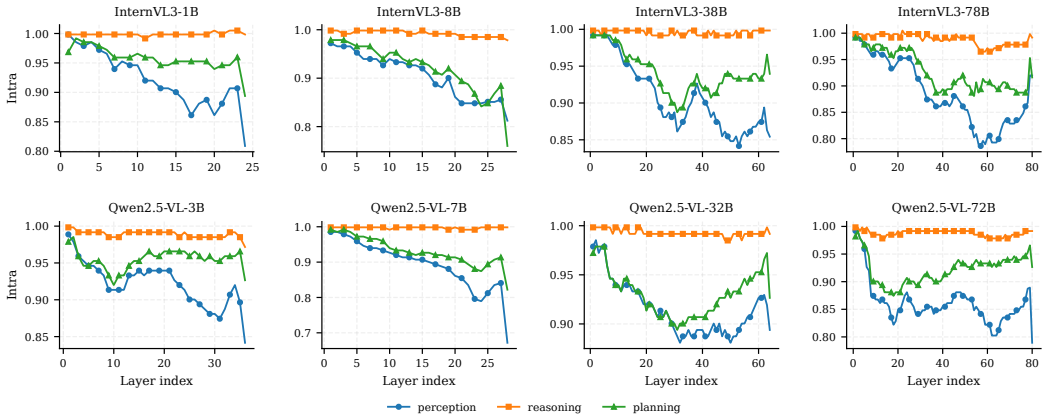


Figure 5: Capability-wise intra-consistency (mean pairwise cosine similarity) across Transformer layers for InternVL3 (top row) and Qwen2.5-VL (bottom row) under three task types: perception, reasoning, and planning.

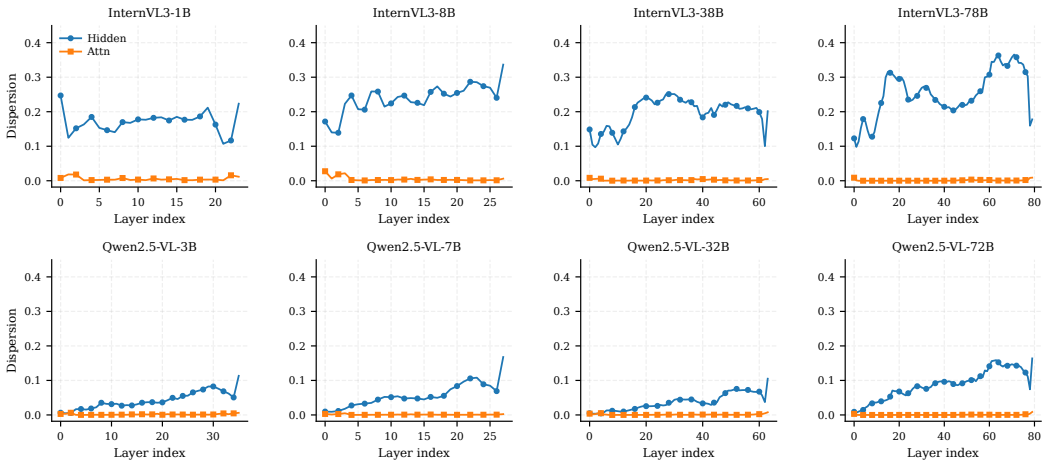


Figure 6: Hidden-state vs. attention dispersion under rewording for InternVL3 and Qwen2.5-VL model families. Each subplot corresponds to one model; the x-axis is the Transformer layer index and the y-axis is dispersion measured as $1 - \cos(\cdot, \cdot)$ on the all-capabilities aggregate. We report dispersion curves for the last-token hidden representation (Hidden) and the head-mean attention map (Attn). Lower values indicate more stable signals across different question rephrasings on the same image.

B.4 Position-wise output-distribution alignment analyses

We present position-wise analyses for output-distribution alignment under multiple temperatures (Figures 7 and 8), contrasting driving and general answers to clarify why KL-based output alignment can become noisy on driving outputs.

C Optimization Details

This section details the online loss reweighting procedure used in training, including how per-loss gradient magnitudes are estimated and how the weights are updated to stabilize multi-objective optimization.

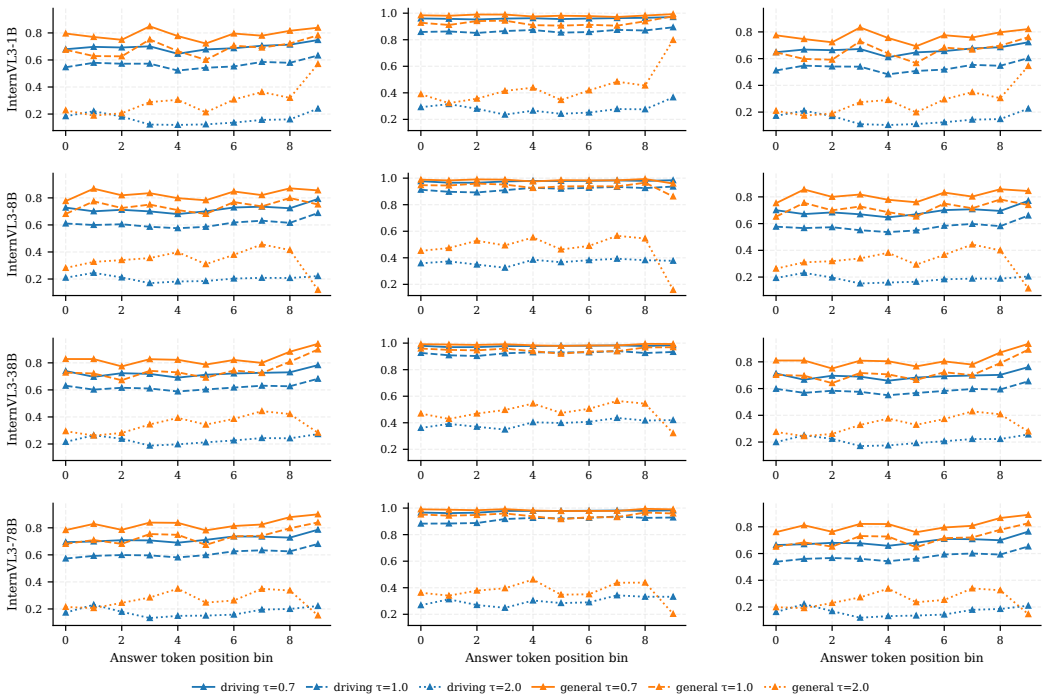


Figure 7: Output-distribution alignment (KL) analysis on answer-token position bins for the InternVL3 family. Rows correspond to models (1B/8B/38B/78B), and columns report three confidence metrics: top-1 probability m, top-10 mass S, and generalized margin. For each subplot, we compare *driving* vs. *general* answers under three temperatures ($\tau \in \{0.7, 1.0, 2.0\}$), using consistent colors (split) and line styles (temperature).

C.1 Online loss reweighting strategy.

We use an online loss reweighting strategy to stabilize multi-objective training, balancing the influence of different losses on parameter updates and reducing sensitivity to manual tuning. Specifically, each loss term L_i is assigned a positive learnable weight w_i, and we jointly optimize the weighted sum $\sum_i w_i L_i$ together with the student model. At each optimization step, we estimate the gradient magnitude of each loss as $G_i = \|\nabla_\theta L_i\|_2$ and update $\{w_i\}$ so that the weighted magnitudes $w_i G_i$ approach a target value. The target combines a preset static weight ratio p_i and the relative decay rate of each loss, where s indexes training steps:

$$\hat{G}_i \propto p_i \left(\frac{L_i(s)/L_i(0)}{\frac{1}{N}\sum_{j\in\mathcal{I}} L_j(s)/L_j(0)} \right)^{\alpha}. \tag{8}$$

We encourage the weighted gradient strengths to satisfy $w_i G_i \approx \hat{G}_i$. Here $\mathcal{I}$ is the set of active loss terms and $N = |\mathcal{I}|$, while α controls how strongly the reweighting reacts to relative loss decay.

C.2 Teacher mixing weights.

In multi-teacher training (Section 3.4), for a batch from capability $c \in \{p, r, pl\}$, we use the fixed teacher-mixing matrix

$$\Pi = \begin{bmatrix} 0.8 & 0.1 & 0.1 \\ 0.1 & 0.8 & 0.1 \\ 0.1 & 0.1 & 0.8 \end{bmatrix}. \tag{9}$$

Each row corresponds to the batch capability (perception/reasoning/planning), and each column corresponds to the perception, reasoning, and planning teachers, where each teacher defines a fixed capability-specific loss L_t independent of the batch capability c. For a batch from capability c, the

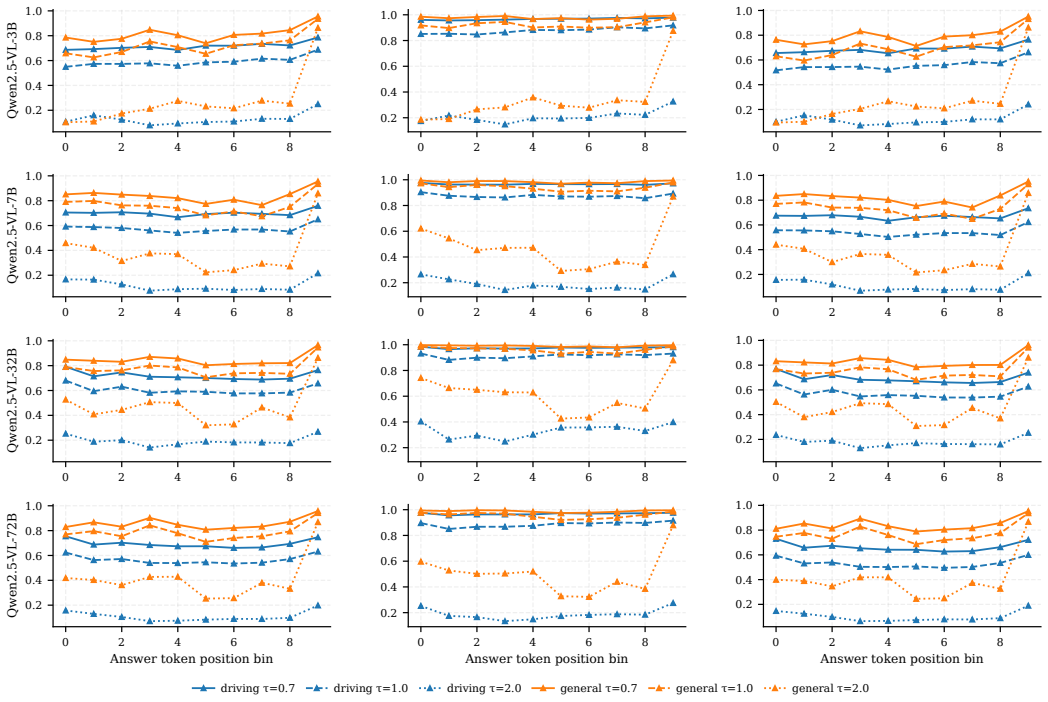


Figure 8: KL-based logit distillation analyses on answer-token position bins for the Qwen2.5-VL family. Rows correspond to models (3B/7B/32B/72B), and columns report three confidence metrics: top-1 probability m, top-10 mass S, and generalized margin. Each subplot contrasts *driving* vs. *general* answers under three temperatures ($\tau \in \{0.7, 1.0, 2.0\}$), with consistent colors (split) and line styles (temperature).

c-th row of Π provides $\pi_{c,t}$ to combine the three teacher losses. This assigns the largest weight to the specialized teacher while keeping non-zero contributions from the other teachers. This reduces overfitting to a single capability and mitigates catastrophic forgetting across capabilities.

D Gradient Conflict Analysis

The two-stage AGP procedure is illustrated in Figure 9.

We directly measure negative cosine events between CE and capability-specific soft gradients and among the three soft gradients during training.

Without gradient treatment, approximately half of the measured gradient pairs conflict in both model families; AGP removes every CE–soft conflict and reduces soft–soft conflicts to below one percent. Table 6 reports the conflict rates before and after AGP.

Table 6: Fraction of measured gradient pairs with negative cosine similarity during multi-teacher training.

Family	Gradient pair	No AGP (%)	AGP (%)
InternVL3	CE–Soft	51.34	**0.00**
InternVL3	Soft–Soft	48.31	**0.83**
InternVL3	All pairs	49.82	**0.42**
Qwen2.5-VL	CE–Soft	54.11	**0.00**
Qwen2.5-VL	Soft–Soft	49.44	**0.56**
Qwen2.5-VL	All pairs	51.77	**0.28**

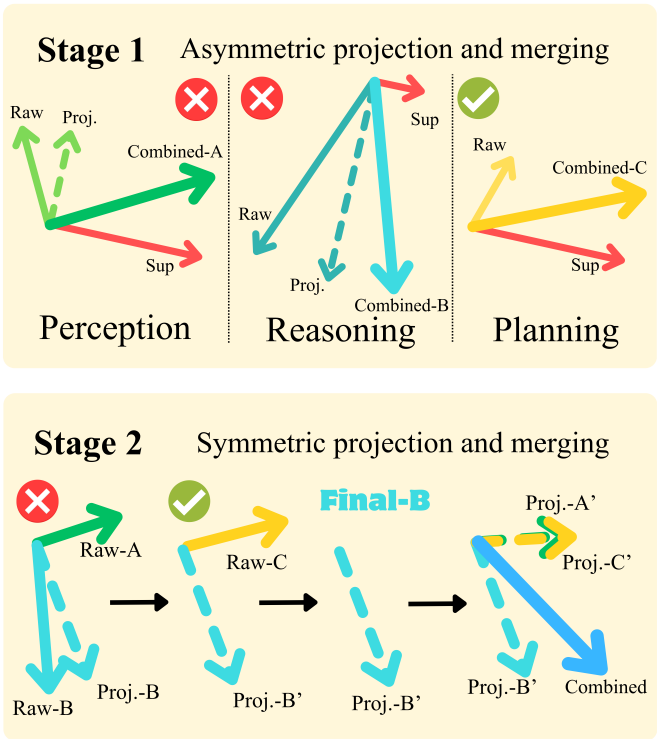


Figure 9: **Asymmetric Gradient Projection (AGP).** Stage 1 uses an asymmetric anchor–follower projection within each capability and merges the update. Stage 2 applies shuffled symmetric pairwise projections across capabilities to obtain the final gradient direction (gradient B shown as an example).

Together with the performance ablations in Section 4.6, these measurements provide direct evidence that gradient conflicts are frequent and that AGP resolves them effectively.

E Annotated Distillation Data

E.1 Dataset Sources and Annotation Format

Multi-view image groups are sampled from nuScenes, and single-view images are sampled from both nuScenes and BDD100K Caesar et al. (2020); Yu et al. (2020a). For each image (single-view) or image group (multi-view), we annotate eight driving-related question types spanning the sequential triad of perception, reasoning, and planning. Each instance is paired with a human reference answer consisting of a brief final response and a detailed chain-of-thought rationale.

E.2 Dataset Diversity Statistics

Our human annotations span eight driving-related question types, including motion status, target identification, multi-view object extraction and attention-based reasoning, ego-behavior reasoning, rationale-grounded planning decisions, safe-action set analysis, and collision-inducing action analysis. The dataset spans three capability categories, and includes both single-image and six-image settings (56.23% vs. 43.77%), where the latter covers surround-view camera perspectives. In addition, targets are broadly distributed in the image space: center-mid 43.10%, right-mid 13.95%, left-mid 11.85%, center-bottom 10.57%, right-bottom 7.41%, left-bottom 4.84%, center-top 4.45%, right-top 2.46%, and left-top 1.37%.

At the object level, for road-structure and traffic-rule related objects, lane accounts for 59.60%, traffic sign 25.90%, and traffic light/traffic signal 9.97%, along with long-tail elements such as crosswalk, curb, sidewalk, and stop sign. For vehicles, sedan accounts for 43.35%, SUV 26.96%,

and truck/van 21.40%, and the annotations also include bus, pickup, and taxi. For vulnerable road users, the annotations cover pedestrians, motorcycles, bicycles, and cyclists. These objects are accompanied by precise color and shape descriptions.

At the scene level, the annotations span diverse road structures and interactions, dominated by turning (50.20%) and intersections (14.21%), followed by lane change (8.74%), parking (8.55%), and crosswalk areas (7.19%), with additional coverage of construction, traffic jams and ramp merges. Weather and time-of-day mentions are also diverse (clear/rain/overcast, plus snow/wind/fog; and night/daytime/dawn–dusk). The action space includes both routine safe decisions and counterfactual risky behaviors: stop (26.69%), keep speed (15.98%), turn right/left (14.63%/14.15%), gentle braking to stop (7.75%), gradual deceleration without braking (4.83%), slight offset left/right (6.53% total), lane change left/right (3.34% total), and collision-oriented counterfactuals such as accelerating straight (3.40%) as well as sustained deviation and sharp turns.

Finally, at the language level, the average annotation length is about 61 words / 410 characters, with a clear spread (p25=48 words, p75=71 words, max=205 words), demonstrating rich variation in expression and explanatory details.

E.3 Annotated Examples

We group representative annotated examples for perception, reasoning, and planning (Figures 10 to 12) to illustrate the format of our 10,500 distillation dataset. Each instance contains a brief answer to the question followed by a detailed chain-of-thought rationale.

Annotated Example (Perception)



Front (single-view)

Question: What is the moving status of object <c1,CAM_FRONT,0.3843, 0.3338>? Please select the correct answer from the following options: A. Going ahead. B. Turn left. C. Turn right.

Answer: B

Rationale: Objects: Black SUV (ID 1). Location & orientation: left of ego at mid-distance; its heading shows a slight tilt to the left, indicating a left turn relative to the lane/curb. Environment: cloudy weather, dry road surface, green traffic signal. Reasoning: the observed trajectory and slight leftward tilt, together with its position in the left lane, indicate it is turning left; lane markings and alignment with the curb further support this motion state.

Figure 10: Perception annotation example from our 10,500 distillation dataset.

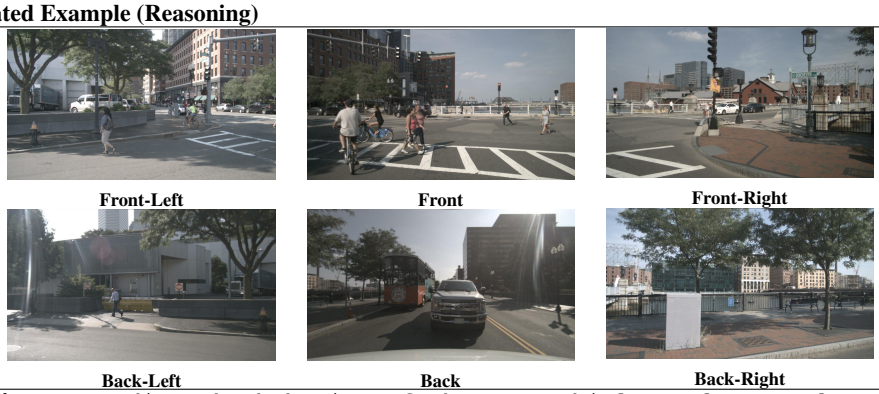
F Training Details

F.1 Reproducibility details.

To facilitate reproducibility, we report the full training configuration used in all main distillation experiments.

Training setup. We use full-parameter fine-tuning with a fixed random seed of 42 on NVIDIA A100-PCIE-40GB GPUs. The software environment uses Ubuntu 22.04.5 LTS, CUDA 12.4, Python 3.10.20, PyTorch 2.6.0+cu124, and Transformers 4.50.3. For the static loss coefficients, we use $w_{\text{ce}} = 0.7$, $w_p = 60$, $w_r = 0.15$, and $w_{pl} = 60$ in the main experiments. The larger perception and

Annotated Example (Reasoning)



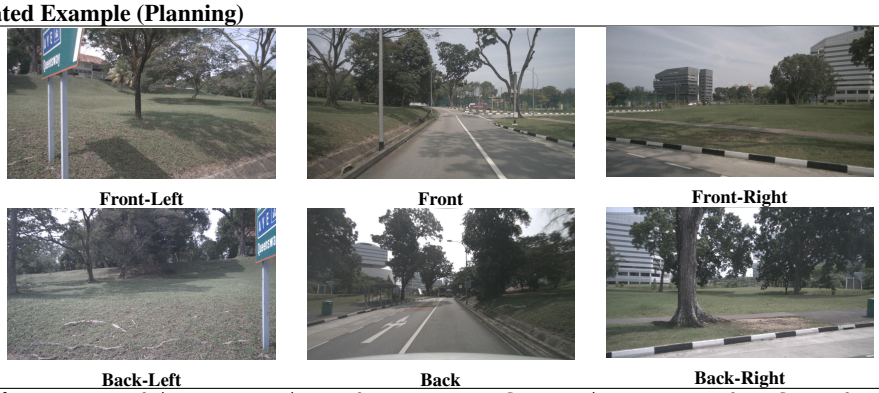
Question: Predict the behavior of the ego vehicle. Please select the correct answer from the following options: A. The ego vehicle is slightly steering to the right, driving with normal speed. B. ... C. ... D. ...

Answer: A

Rationale: The vehicle is traveling at the intersection of two straight roads. Object <own vehicle> is located in the middle of the road, facing directly forward relative to the camera. Based on the road markings and the vehicle's orientation in the scene, its behavior is judged as **A**. The weather is clear, which does not alter the judgment.

Figure 11: Reasoning annotation example from our 10,500 distillation dataset.

Annotated Example (Planning)



Question: In this scenario, what are safe actions to take for the ego vehicle?

Answer: keep going at the same speed; decelerate gradually without braking

Rationale: The road ahead is clear with no immediate obstacles or traffic, and the lane markings indicate a straight path. The presence of a bus in the distance suggests normal traffic flow, so maintaining speed or gently reducing it is appropriate.

Figure 12: Planning annotation example from our 10,500 distillation dataset.

planning weights are used because their supervision signals are sparser and require stronger scaling for stable convergence. Each reported model result is based on three training runs, each lasting 3 epochs on our 10,500 human-annotated driving distillation dataset. The SFT baselines in the open-loop and closed-loop main tables are unified CE-only models trained jointly on the complete multi-capability training set of the corresponding experiment.

Optimization hyperparameters. The student model is optimized with AdamW. We use a learning rate of 5×10^{-5}. We use a cosine learning-rate scheduler with a warmup ratio of 0.03 and a minimum learning-rate ratio of 0.01. We set the micro-batch size to 1, and use gradient accumulation with 4 steps, resulting in an effective batch size of $1 \times 4 = 4$.

Online loss reweighting implementation. In Appendix C.1, we parameterize each learnable loss weight as $w_i = \mathrm{softplus}(a_i) + \epsilon$ with $\epsilon = 10^{-8}$. We then apply a bounded constraint to w_i and use small positive epsilons for numerical stability. The dynamic-weight parameters $\{a_i\}$ are optimized with AdamW using learning rate 3×10^{-4}, with standard numerical-stability safeguards.

AGP numerical stability. In the projection operator in Section 3.4, we add a small denominator constant $\epsilon_{\mathrm{proj}} = 10^{-20}$ when computing $\|\mathbf{g}\|_2^2$. This prevents division-by-zero in degenerate cases.

G Evaluation Details and Additional Results

G.1 Benchmark protocols and evaluation settings

We evaluate open-loop driving capabilities on DriveBench and Drive-P2D, and closed-loop driving on Bench2Drive-VL. For DriveBench, we merge its prediction and behavior categories into reasoning and use DeepSeek-V3.2 as the judge.

G.1.1 DriveBench inference metrics

Both training and inference are conducted on NVIDIA A100-PCIE-40GB GPUs. Inference metrics are measured with the minimum number of GPUs required to run each model successfully. All inference measurements are conducted on DriveBench. We report three deployment-oriented inference metrics: peak GPU memory, average generation throughput, and median time-to-first-token (TTFT).

- **Peak GPU memory (GB).** We run inference with the minimum number of GPUs that can successfully execute the model, and report peak GPU memory as the maximum total memory usage summed over all GPUs observed during the run.
- **Average generation throughput (tokens/s).** Average generation throughput is defined as the number of newly generated tokens divided by the total generation time:

$$\mathrm{avg_gen_tokens_per_s} = \frac{\mathrm{total_gen_tokens}}{\mathrm{gen_time_total_s}}, \tag{10}$$

  where $\mathrm{total_gen_tokens}$ excludes input prompt tokens.

- **Median TTFT (seconds).** We measure TTFT by repeatedly running generation with $\mathrm{max_new_tokens} = 1$ and recording the elapsed time to produce the first token. We report the median over runs:

$$\mathrm{ttft_s_median} = \mathrm{median}\left(\{T_i\}\right). \tag{11}$$

G.1.2 DriveBench grading protocol

DriveBench uses an LLM-based grader with predefined rules and scene metadata. In our evaluation, we keep the grading prompt fixed and run grading five times, reporting the average score to reduce randomness. We quantify the stochasticity of LLM-based grading by reporting the run-to-run standard deviation of the aggregated capability scores over the five independent grading runs.

Table 7: Run-to-run standard deviation σ (%) of DriveBench capability scores over five DeepSeek-V3.2 grading runs for the models in Table 1.

Model	Perception σ	Reasoning σ	Planning σ
GPT-5.1	1.1787	1.5382	1.4031
InternVL3-1B	0.9494	1.9117	1.6508
InternVL3-2B	1.3180	1.5707	1.4936
InternVL3-8B	0.8428	1.6398	1.5451
InternVL3-14B	0.8410	1.6294	1.6342
InternVL3-38B	1.1267	1.6408	1.5969
InternVL3-78B	1.2724	1.5285	1.7304
Qwen2.5-VL-3B-Instruct	1.2111	1.7037	1.3293
Qwen2.5-VL-7B-Instruct	1.3538	1.7018	1.2906
Qwen2.5-VL-32B-Instruct	0.8797	1.9699	1.6058
Qwen2.5-VL-72B-Instruct	1.2711	1.9081	1.5641
Llama-3.2-11B-Vision-Instruct	1.3626	1.7238	1.5739
Llama-3.2-90B-Vision-Instruct	1.2834	1.8497	1.7649
InternVL3-1B (Single)	1.1445	1.8949	1.2617
Qwen2.5-VL-3B-Instruct (Single)	0.9520	1.6770	1.2826
InternVL3-1B (Multi)	0.9547	1.5781	1.4049
Qwen2.5-VL-3B-Instruct (Multi)	1.1767	1.7166	1.4617

Specifically, for each model and capability, let $\{s_i\}_{i=1}^{5}$ be the overall DriveBench score (%) from the i-th grading run and $\bar{s} = \frac{1}{5}\sum_{i=1}^{5} s_i$ be the mean. We report the sample standard deviation

$$\sigma = \sqrt{\frac{1}{5-1}\sum_{i=1}^{5}(s_i - \bar{s})^2}. \tag{12}$$

Table 7 lists σ for all models in Table 1.

DriveBench evaluation examples. We provide concrete evaluation-case illustrations, clarifying how our protocol is applied in practice. Specifically, Figures 14 to 16 present representative cases for perception, reasoning, and planning, respectively.

Drive-P2D. For Drive-P2D, we follow its evaluation protocol and report object perception, scene understanding, decision making, and their average.

Bench2Drive-VL. For Bench2Drive-VL, we follow the same inference setup as the authors' and evaluate Base, SFT, and Drive-KD on the same ten Dev10 routes with front-camera input, reporting Driving Score and Route Completion.

G.2 DRIVE-P2D RESULTS

We further evaluate InternVL3-1B on Drive-P2D Tang et al. (2026), an autonomous-driving benchmark spanning perception, scene understanding, and decision making. We compare the base model, single-capability SFT models, single-capability distillation models, and the multi-teacher Drive-KD model. Table 8 reports the corresponding results on Drive-P2D.

Table 8: Drive-P2D results for InternVL3-1B variants (%).

Model	Obj.	Scn.	Dec.	Avg.
Base	39.85	12.83	29.39	27.36
SFT-Perc.	45.80	40.96	28.24	38.33
Single-Perc.	48.73	46.17	26.96	40.62
SFT-Reas.	40.30	20.42	27.94	29.55
Single-Reas.	44.19	17.96	30.63	30.93
SFT-Plan.	43.74	27.79	37.89	36.47
Single-Plan.	46.11	31.25	40.18	39.18
Multi	**47.67**	**46.81**	**41.45**	**45.31**

Consistent with DriveBench, distillation generally improves over the corresponding SFT baselines under single-capability training, and multi-teacher distillation gives the best average score. These results support the effectiveness of Drive-KD.

G.3 Comparison with representative KD methods

We compare Drive-KD with three recent VLM/MLLM distillation baselines, EPIC/PCD Wen et al. (2025), Align-KD Feng et al. (2024), and Beta-KD Sun et al. (2026). EPIC/PCD uses progressive consistency distillation for efficient multimodal large language models, Align-KD transfers cross-modal alignment knowledge from a larger VLM to a smaller one, and Beta-KD adaptively reweights distillation supervision according to uncertainty. We adapt these baselines only where required for model compatibility, train them on the same autonomous-driving data, and evaluate them with the same protocol. Table 9 reports the comparison results under this setting.

Table 9: Comparison with representative distillation baselines on InternVL3-1B using DriveBench (%).

Model	Perception	Reasoning	Planning	Avg.
InternVL3-1B	33.26	20.96	22.36	25.53
Ours (InternVL3-1B, Single)	43.13	**34.32**	52.97	43.47
Ours (InternVL3-1B, Multi)	**43.50**	33.15	**55.51**	**44.05**
EPIC/PCD (InternVL3-1B)	38.76	32.56	33.77	35.03
Align-KD (InternVL3-1B)	37.31	30.19	31.20	32.90
Beta-KD (InternVL3-1B)	38.14	33.73	39.95	37.27

Drive-KD obtains the best overall performance under the same driving setting, indicating stronger transfer of driving-specific capabilities than these representative KD baselines.

H Capability Alignment and Visual Attention

H.1 Reasoning–planning alignment

We examine whether stronger reasoning performance on a scene is associated with stronger planning performance for the final Drive-KD students and GPT-5.1. For each image, we average its reasoning-question scores and planning-question scores, and then compute Pearson and Spearman correlations across images containing both question types. Table 10 reports the resulting image-level correlations.

Table 10: Image-level correlation between reasoning and planning scores.

Model	Pearson	Spearman
GPT-5.1	-0.0418	-0.0316
Drive-KD (InternVL3)	$\mathbf{+0.3384}$	$\mathbf{+0.3724}$
Drive-KD (Qwen2.5-VL)	$+0.2976$	$+0.3271$

Drive-KD produces clear positive reasoning–planning alignment in both model families, whereas GPT-5.1 is near zero, showing that our student models form a more coherent same-scene mapping from reasoning to action planning.

H.2 Visual grounding with Drive-KD

We quantify answer-token attention on the same 600 held-out Bench2Drive-VL VQA records for Base, SFT, and Drive-KD in each model family, with 150 records sampled from each original benchmark category. Here, SFT denotes the unified CE-only model trained jointly on our complete multi-capability training set. Visual attention mass measures how strongly answer tokens attend to visual tokens, while Top-10% concentration and normalized entropy characterize selective focus versus broader evidence integration. Table 11 reports these metrics for the three driving capabilities.

An additional qualitative example is shown in Figure 13. The Drive-KD–SFT changes reveal the

Table 11: Answer-token visual grounding across the three driving capabilities. InternVL3 and Qwen2.5-VL denote InternVL3-1B and Qwen2.5-VL-3B-Instruct, respectively. Attention mass measures visual grounding strength, while concentration and entropy reveal capability-specific evidence organization.

	Visual attention mass (%)			Top-10% conc. (%)			Normalized entropy		
Model	Perception	Reasoning	Planning	Perception	Reasoning	Planning	Perception	Reasoning	Planning
InternVL3	14.47	7.37	10.40	70.24	66.63	76.89	0.775	0.795	0.752
InternVL3 (SFT)	4.26	0.34	0.25	57.05	56.04	73.01	0.860	0.743	0.562
InternVL3 (Ours)	4.95	3.11	3.09	63.07	45.83	43.85	0.827	0.907	0.910
Qwen2.5-VL	2.12	5.91	8.34	58.84	49.92	55.70	0.842	0.868	0.843
Qwen2.5-VL (SFT)	0.90	0.38	0.34	18.11	62.84	93.22	0.990	0.719	0.507
Qwen2.5-VL (Ours)	2.96	4.33	5.33	30.38	45.53	50.08	0.955	0.900	0.873

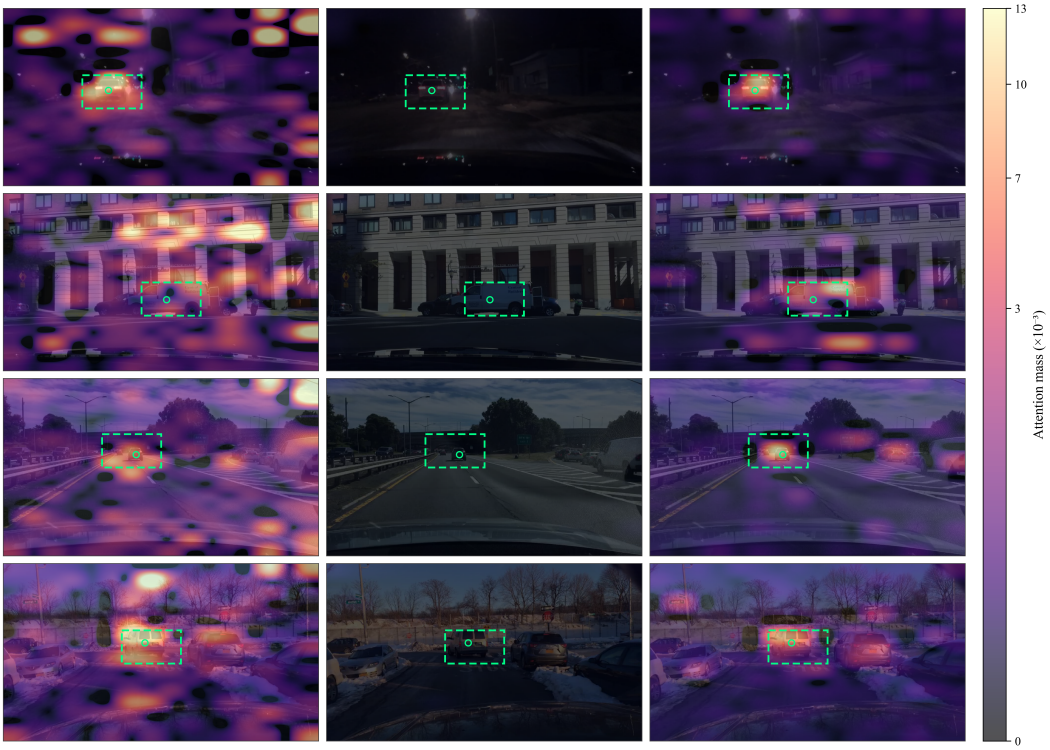


Figure 13: **Additional visual-grounding example.** From left to right: Base, SFT, and Drive-KD, visualized with a shared attention scale.

same ordered attention pattern in both model families. For perception, Drive-KD raises Top-10% concentration by 6.02 and 12.27 points and lowers entropy by 0.033 and 0.035 for InternVL3 and Qwen2.5-VL, respectively, sharpening attention around selective visual cues. For reasoning, concentration decreases by 10.21 and 17.31 points while entropy rises by 0.164 and 0.181, expanding attention toward relational context. Planning shows the strongest redistribution: concentration decreases by 29.16 and 43.14 points while entropy rises by 0.348 and 0.366, enabling broad scene-level evidence integration.

SFT substantially suppresses visual grounding relative to Base in every capability: visual attention mass falls from 14.47/7.37/10.40 to 4.26/0.34/0.25 for InternVL3 and from 2.12/5.91/8.34 to 0.90/0.38/0.34 for Qwen2.5-VL across perception/reasoning/planning. Drive-KD restores this mass to 4.95/3.11/3.09 and 2.96/4.33/5.33, respectively, with the strongest recovery in reasoning and planning.

Table 13: Per-capability layer ablations for InternVL3-1B single-teacher distillation on DriveBench (%).

Setting	P	R	Pl
CE + $A_{t\to v}(1)$	**43.13**	–	–
CE + $A_{t\to v}(2)$	41.81	–	–
CE + $A_{t\to v}(3)$	41.84	–	–
CE + $A_{t\to v}(\ell_{\text{pen}})$	40.85	–	–
CE + FA-mid $(1 \to \ell_{\text{pen}})$	–	**34.32**	–
CE + FA-mid $(2 \to \ell_{\text{pen}} - 1)$	–	31.43	–
CE + FA-mid $(3 \to \ell_{\text{pen}} - 2)$	–	31.22	–
CE + $A_{t\to v}(\ell_{\text{pen}} - 1)$	–	–	51.16
CE + $A_{t\to v}(\ell_{\text{pen}})$	–	–	**52.97**
CE + $A_{t\to v}(N)$	–	–	51.50
CE + $A_{t\to v}(1)$	–	–	49.36

We further test whether this restored attention reaches coordinate-defined target regions on 48 fixed single-front-view planning records from our open-loop driving QA data. Each normalized object coordinate in the QA annotation is mapped to the visual-token grid, where its centered 3×3 neighborhood defines the target patch. Table 12 reports visual attention mass, the share assigned to target patches, and the frequency with which the most-attended patch falls inside the target region.

Table 12: Coordinate-grounded answer-token visual attention on 48 planning records. InternVL3 and Qwen2.5-VL denote InternVL3-1B and Qwen2.5-VL-3B-Instruct, respectively.

Model	Visual mass (%)	Target share (%)	Top-1 hit (%)
InternVL3	**30.16**	11.96	16.7
InternVL3 (SFT)	0.09	2.09	0.0
InternVL3 (Ours)	5.86	**18.27**	**41.7**
Qwen2.5-VL	**15.58**	2.00	0.0
Qwen2.5-VL (SFT)	0.10	0.04	0.0
Qwen2.5-VL (Ours)	4.87	**9.38**	**25.0**

Drive-KD assigns the largest target-patch share in both families and raises the target top-1 hit rate to 41.7% for InternVL3 and 25.0% for Qwen2.5-VL, compared with 0% after SFT. Together, the capability analysis and coordinate-grounded results show that Drive-KD not only restores visual attention but organizes it around the evidence required by perception, reasoning, and planning.

I Additional Ablation Details

I.1 Per-capability layer ablations

We further evaluate per-capability single-teacher distillation across different layer choices on InternVL3-1B. Each row trains on the corresponding capability split and reports the score of that capability, using the same DriveBench evaluation protocol as the main experiments. Table 13 reports the resulting per-capability layer ablations.

The best perception result is obtained by early text-to-vision attention at Layer 1, while planning is strongest at the penultimate layer. For reasoning, the broad intermediate full-attention range performs best, supporting the design that reasoning should connect early perception cues with later decision-related features.

I.2 Additional attention distillation variants

This appendix reports supplementary ablation results. We compare two types of variants against our final single-teacher recipe. First, we test alternative attention targets that deviate from our recipe:

Table 14: InternVL3-1B extended comparisons on DriveBench (%). Each row targets one capability; other entries are not applicable ("–"). For Ours (single-teacher), each reported score is obtained by distilling and evaluating that capability using its corresponding data only, consistent with the single-capability setting above.

Variant	Perception	Reasoning	Planning
CE + Full Attn (1)	42.46	–	–
CE + $A_{t\text{–}v}$ (mid)	–	30.42	–
CE + Full Attn (ℓ_{pen})	–	–	51.47
CE + $A_{t\text{–}v}$ (1), cosine	41.85	–	–
CE + Full Attn (mid), cosine	–	32.87	–
CE + $A_{t\text{–}v}$ (ℓ_{pen}), cosine	–	–	51.76
Ours (single-teacher)	**43.13**	**34.32**	**52.97**
Ours (multi-teacher)	**43.50**	**33.15**	**55.51**

using full attention at Layer 1 for perception, using text-to-vision attention ($A_{t\text{–}v}$) in intermediate layers for reasoning, and using full attention at ℓ_{pen} for planning. Second, we test similarity-based loss as a replacement for MSE. Table 14 reports the corresponding extended comparisons.

We consider two attention targets: (i) full attention, which matches all query–key entries of the head-mean attention matrix, and (ii) text-to-vision attention $A_{t\text{–}v}$, which matches the submatrix whose queries are text tokens and keys are vision tokens. In all attention-based distillation losses, we apply the standard attention masks (causal and padding) and compute the objective only on valid query–key positions shared by the teacher and student sequences. For $A_{t\text{–}v}$, we define T_b as the set of all text-query token positions in the concatenated prompt+answer sequence (excluding special and padded tokens), and define V_b as the set of all vision-key token positions. For similarity-based loss, we define cosine-based attention matching as a replacement for MSE.

Perception / planning (single-layer $A_{t\text{–}v}$ cosine loss). We apply a cosine-based loss to the text-to-vision attention $A_{t\text{–}v}$ at Layer 1 and ℓ_{pen}, respectively. For sample b, let T_b be the index set of text-query positions and V_b be the index set of vision-key positions. Define the vectorization operator $\text{vec}_{t\text{–}v}(\bar{A}^{(\ell)}_{(\cdot),b}) \in \mathbb{R}^{|T_b||V_b|}$ that stacks the entries $\{\bar{A}^{(\ell)}_{(\cdot),b,i,j}\}_{i\in T_b,\, j\in V_b}$. We then apply L_2 normalization and compute cosine dissimilarity:

$$\begin{aligned}\widetilde{\mathbf{a}}^{(\ell)}_{(\cdot),b,t\text{–}v} &= \text{vec}_{t\text{–}v}\left(\bar{A}^{(\ell)}_{(\cdot),b}\right),\\ \widehat{\mathbf{a}}^{(\ell)}_{(\cdot),b,t\text{–}v} &= \frac{\widetilde{\mathbf{a}}^{(\ell)}_{(\cdot),b,t\text{–}v}}{\left\|\widetilde{\mathbf{a}}^{(\ell)}_{(\cdot),b,t\text{–}v}\right\|_2 + \epsilon},\end{aligned} \tag{13}$$

where ϵ is a small constant for numerical stability. The cosine loss is

$$\mathcal{L}^{\text{sim-}t\text{–}v}_{\text{att}}(\ell) = \text{mean}_b\left[1 - \cos\left(\widehat{\mathbf{a}}^{(\ell)}_{s,b,t\text{–}v}, \widehat{\mathbf{a}}^{(\ell)}_{t,b,t\text{–}v}\right)\right], \tag{14}$$

where $\cos(\mathbf{u},\mathbf{v}) = \mathbf{u}^\top\mathbf{v}$ for L_2-normalized vectors.

Reasoning (grouped intermediate layers; full-attention cosine loss). For intermediate-layer reasoning supervision, teacher and student may have different depths. We therefore reuse the layer-group matching in Section 3.3: let the selected student layer set be $\mathcal{S}$ and map each $\ell \in \mathcal{S}$ to a nearby teacher-layer group $\mathcal{G}(\ell)$. We form the teacher target as the group-mean head-mean attention:

$$\mu^{(\ell)}_{t,b,i,j} = \text{mean}_{k\in\mathcal{G}(\ell)}\, \bar{A}^{(k)}_{t,b,i,j}. \tag{15}$$

Let $\text{vec}_{\text{full}}(\cdot)$ stack all entries of a head-mean full attention matrix. We define the student/teacher attention vectors and apply L_2 normalization:

$$\widetilde{\mathbf{a}}^{(\ell)}_{s,b,\text{full}} = \text{vec}_{\text{full}}\left(\bar{A}^{(\ell)}_{s,b}\right), \qquad \widetilde{\mathbf{a}}^{(\ell)}_{t,b,\text{full}} = \text{vec}_{\text{full}}\left(\mu^{(\ell)}_{t,b}\right), \tag{16}$$

$$\widehat{\mathbf{a}}^{(\ell)}_{(\cdot),b,\text{full}} = \frac{\widetilde{\mathbf{a}}^{(\ell)}_{(\cdot),b,\text{full}}}{\left\|\widetilde{\mathbf{a}}^{(\ell)}_{(\cdot),b,\text{full}}\right\|_2 + \epsilon}. \tag{17}$$

The grouped cosine distillation loss is

$$\mathcal{L}^{\text{sim-full}}_{\text{att}} = \text{mean}_b \left[\frac{1}{|\mathcal{S}|} \sum_{\ell \in \mathcal{S}} \left(1 - \cos\left(\widehat{\mathbf{a}}^{(\ell)}_{s,b,\text{full}}, \widehat{\mathbf{a}}^{(\ell)}_{t,b,\text{full}}\right)\right) \right]. \tag{18}$$

As shown in Table 14, we find that alternative attention choices are consistently weaker than our final single-teacher recipe on the targeted capability. Finally, MSE attention matching is generally stronger than similarity-based matching in this setting, so we use MSE as the default attention distillation loss throughout Drive-KD.

J Existing Assets and Terms

Datasets and benchmarks. Our 10,500 driving distillation dataset is derived from images sampled from nuScenes Caesar et al. (2020) and BDD100K Yu et al. (2020a). The relevant upstream terms are the nuScenes Terms of Use and the BDD100K BSD 3-Clause license. DriveBench Xie et al. (2025) is our main evaluation benchmark; we use its public main branch, commit range `5eb807b--5f9ac57`, under Apache-2.0 with upstream terms applying where relevant. Drive-P2D Tang et al. (2026) is used as an additional autonomous-driving benchmark, with its upstream terms applying where relevant. Bench2Drive-VL Jia et al. (2026) is used for closed-loop evaluation; its released VQA data are used for closed-loop adaptation and open-loop visual-attention analysis. Vision-Flan Xu et al. (2024b), specifically `vision-flan_191-task_1k`, is used as generic multimodal QA data for the pre-study analysis; its underlying source-dataset licenses apply.

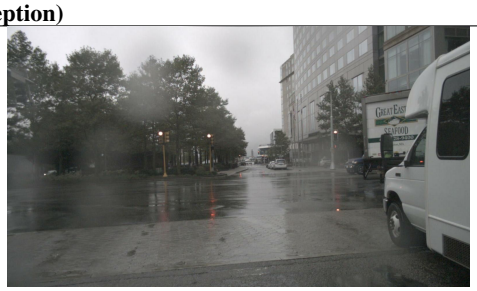
Pretrained and API models. We use InternVL3 checkpoints from OpenGVLab Chen et al. (2024c) as students, teachers, scaling variants, and baselines. The InternVL project is MIT-licensed, and the license of the Qwen2.5 component applies where relevant. We use Qwen2.5-VL checkpoints Team (2025) as students, teachers, and baselines; the 7B and 32B checkpoints are under Apache-2.0, while the 3B checkpoint is under the Qwen Research License Agreement and the 72B checkpoint is under the Qwen License Agreement. We use Llama-3.2-11B-Vision-Instruct and Llama-3.2-90B-Vision-Instruct as baselines under the Llama 3.2 Community License and Acceptable Use Policy. DeepSeek-V3.2 DeepSeek-AI et al. (2025) is used through the official API as the LLM-based evaluator under the DeepSeek Open Platform Terms of Service. GPT-5.1 is used as a proprietary API baseline through OpenRouter model ID `openai/gpt-5.1`, subject to OpenRouter terms and applicable provider or model terms. We also use the publicly released ReasonDrive baseline model weights `ac4462/Qwen2.5-VL-3B-DriveLM` from Hugging Face under the Apache-2.0 license for comparison only, without redistributing the weights.

Baseline code. We compare with EPIC/PCD Wen et al. (2025) using the official repository at commit `b2ed9cd` under Apache-2.0. We also compare with Align-KD Feng et al. (2024) using the official repository at commit `cfb020d`; no explicit upstream license was specified at the time of use, so we use it only for internal research comparison, do not redistribute code derived from it, and credit the original work. We compare with Beta-KD Sun et al. (2026) using its official repository at commit `634a631`; the repository does not specify an explicit license.

K Limitations

The current framework is limited to image-and-text inputs and does not incorporate other sensor modalities such as video or point clouds, resulting in underutilized temporal and geometric information. Transferring complex multi-step reasoning to smaller models remains a bottleneck, as reflected in the capability scores, and some high-level reasoning patterns are difficult to preserve under limited capacity. Although our study includes both open-loop and closed-loop evaluation, broader real-world validation remains future work.

Evaluation Example (Perception)



CAM_FRONT

Question: What is the moving status of object <c3,CAM_FRONT,0.4755, 0.5231>? Please select the correct answer from the following options: A. Turn right. B. Turn left. C. Going ahead.

Model: GPT-5.1
Prediction (compressed):
Answer: C. Going ahead.
Explanation: Object c3 is the silver van located roughly in the center lane ahead. Its orientation aligns with the lane's straight arrow markings, and its motion appears to follow the lane forward rather than turning left or right....

Model: Ours (Drive-KD)
Prediction:
Final: A. Turn right.
Object: gray van
Position & orientation: relative to ego front, heading turning right
Environment: visible intersection with traffic signals and lane markings
Contextual evidence: the van's trajectory follows the right-turn lane, and its front wheels are angled right, ...

Evaluator: DeepSeek-V3.2
Ground Truth: A
DeepSeek grading for GPT-5.1:
1. Answer Correctness (50 points): 0
2. Object Recognition (10 points): 5
3. Object Location and Orientation (15 points): 5
4. Environmental Condition Awareness (15 points): 0
5. Clarity of Reasoning (10 points): 8
Total Score: 18
Brief Summary: ...
DeepSeek grading for Ours (Drive-KD):
1. Answer Correctness (50 points): 50
2. Object Recognition (10 points): 5
3. Object Location and Orientation (15 points): 10
4. Environmental Condition Awareness (15 points): 10
5. Clarity of Reasoning (10 points): 8
Total Score: 83
Brief Summary: The predicted answer correctly identifies the moving status as ``Turn right'' (A), matching the ground truth, earning full correctness points. Object recognition is partially accurate: the prediction describes a ``gray van,'' but the ground truth specifies a ``white sedan,'' leading to a deduction for misidentification of color and vehicle type, though both are vehicles in a similar context. Location and orientation are well described with details like ``relative to ego front, heading turning right'' and mention of lane markings and trajectory, ...

Figure 14: Illustration of our evaluation process on a DriveBench perception question. For readability, the model responses and DeepSeek justifications are compressed while preserving the grading structure and key reasons.

Evaluation Example (Reasoning)



Question: Predict the behavior of the ego vehicle. Please select the correct answer from the following options: A. The ego vehicle is going straight. The ego vehicle is not moving. B. The ego vehicle is steering to the right. C. ... D. ....

Model: GPT-5.1
Prediction (compressed):
Answer: B.
- In the front view, the upcoming lane curves to the right, and the ego vehicle appears aligned with that curve, indicating a right steering input.
- Several pedestrians are very close to the vehicle, and another car is close behind in the back view....

Model: Ours (Drive-KD)
Prediction:
Final: A.
Support: The ego vehicle is positioned in the center of the lane with no visible steering or trajectory deviation; the road ahead is clear and straight, and there are no indications of motion blur or speed change in the surrounding environment....

Evaluator: DeepSeek-V3.2
Ground Truth: A
DeepSeek grading for GPT-5.1:
1. Answer Correctness (50 points): 0
2. Behavioral Understanding and Detail (15 points): 0
3. Reasoning and Justification (15 points): 0
4. Contextual Relevance (10 points): 0
5. Clarity and Grammar (10 points): 10
Total Score: 10
Brief Summary: ...
DeepSeek grading for Ours (Drive-KD):
1. Answer Correctness (50 points): 50
2. Behavioral Understanding and Detail (15 points): 15
3. Reasoning and Justification (15 points): 15
4. Contextual Relevance (10 points): 10
5. Clarity and Grammar (10 points): 10
Total Score: 100
Brief Summary: The predicted answer exactly matches the correct answer (A), earning full points for correctness. The explanation accurately captures the ego vehicle's behavior of going straight and not moving, with clear details about lane positioning, lack of steering deviation, and environmental cues like a clear road and no motion blur. Reasoning is logical and well-supported ...

Figure 15: Illustration of our evaluation process on a DriveBench reasoning question. For readability, the model responses and DeepSeek justifications are compressed while preserving the grading structure and key reasons.

Evaluation Example (Planning)



CAM_BACK

Question: What actions could the ego vehicle take based on <c1,CAM_BACK,0.5073,0.5778>? Why take this action and what's the probability?

Model: GPT-5.1
Prediction (compressed):
Object:
<c1,CAM_BACK,0.5073,0.5778> (passenger car), behind the ego and closing.
Action: maintain current lane; keep speed smooth; avoid hard braking.
Reason: close follower → reduce rear-end risk; suggests cautious anticipation for potential cross-traffic / distant vehicles...
Probability: 0.8.
Final: maintain lane with smooth, anticipatory speed control.

Model: Ours (Drive-KD)
Prediction:
Object:
<c1,CAM_BACK,0.5073,0.5778> (gray sedan), relative to ego: back-left.
Action: keep going at the same speed
Reason: The gray sedan is in the adjacent lane moving straight ahead with no indication of braking or lane change; the road is clear ahead and no obstacles or hazards are visible in the immediate vicinity.
Probability: high
Final: keep going at the same speed to maintain safe clearance and traffic flow

Evaluator: DeepSeek-V3.2
Ground Truth: keep going at the same speed; no safety issue; probability high.
DeepSeek grading for GPT-5.1:
1. Action Prediction Accuracy (40 points): 30
2. Reasoning and Justification (20 points): 12
3. Probability or Confidence Level (15 points): 10
4. Contextual Awareness and Safety Considerations (15 points): 10
5. Conciseness and Clarity (10 points): 8
Total Score: 70
Brief Summary: ...
DeepSeek grading for Ours (Drive-KD):
1. Action Prediction Accuracy (40 points): 40
2. Reasoning and Justification (20 points): 20
3. Probability or Confidence Level (15 points): 15
4. Contextual Awareness and Safety Considerations (15 points): 15
5. Conciseness and Clarity (10 points): 10
Total Score: 100
Brief Summary: The predicted answer correctly identifies the action as ``keep going at the same speed,'' matching the ground truth exactly. The reasoning is clear, logical, and aligns with safety considerations, noting the absence of immediate hazards and the sedan's stable behavior in an adjacent lane. The probability is appropriately stated as ``high,'' consistent with the ground truth's ``high'' probability. ...

Figure 16: Illustration of our evaluation process on a DriveBench planning question. For readability, the model responses and DeepSeek justifications are compressed while preserving the grading structure and key reasons.