Rethinking Language's Role in Efficient VLA for Autonomous Vehicles: Toward Smarter, Trustworthy Driving

Tongfei Guo[1] **and Lili Su**[1*]

Abstract

Vision–Language–Action (VLA) models are reshaping autonomous driving (AD) by unifying perception, reasoning, and control through language, enabling semantic grounding, interpretable decisions, and better long-tail generalization. But language is expensive onboard: latency and memory budgets are tight, and autoregressive decoding is inherently sequential. We reframe the central question as *when and where* language should act at inference, since inference cost recurs at every deployed frame while training cost is paid once. We introduce the *Language Residue taxonomy* to organize methods by their inference-time use of language: train-time-only supervision (L1), latent non-textual reasoning (L2), conditional invocation (L3), and full per-frame generation (L4). We review representative methods and tag each across five deployment axes (Latency↓, Parameters↓, Memory↓, FLOPs↓, Tokens↓), analyzing them on major open- and closed-loop driving benchmarks (e.g., nuScenes, NAVSIM, Bench2Drive). We further trace how efficient methods from NLP/LLM are adapted in AD, identifying the constraints and motivations driving these adaptations. A continuously updated repository will be available at GitHub.

1 Introduction

Vision-Language-Action (VLA) models have emerged as a compelling paradigm for autonomous driving (AD), extending end-to-end autonomous driving with contextual reasoning (Sima et al., 2024; Xu et al., 2024c; Tian et al., 2024; Jiang et al., 2024b; Shao et al., 2024). The promise is clear: language can mitigate the uncertainties in long-tail scenarios, and align driving behavior with passenger preferences (e.g. safety vs. comfort vs. speeds). Deploying language under hard deadlines exposes a trade-off between its semantic leverage and its inference cost: AD control loops must close within roughly 20–100 ms (10–50 Hz), yet one autoregressive VLM inference on typical onboard hardware costs 500–2000 ms (Huang et al., 2026).

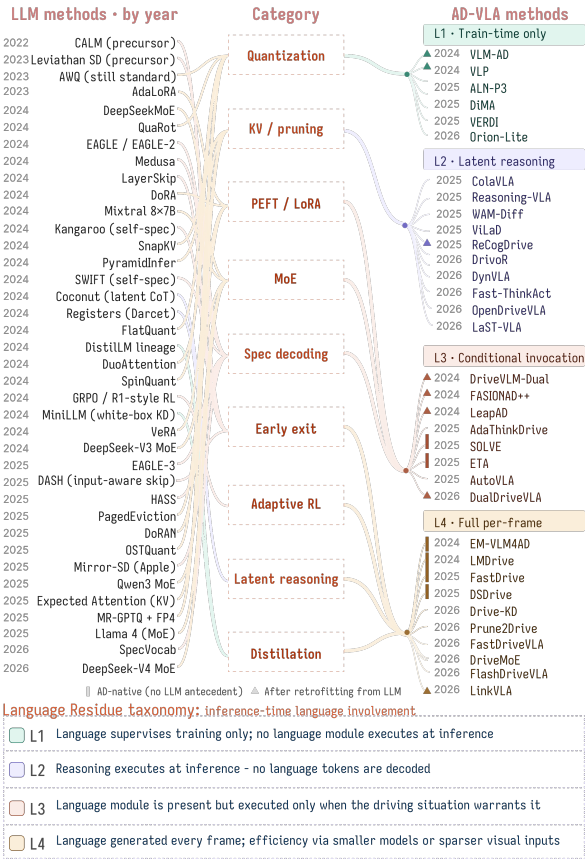

Figure 1: NLP/LLM efficiency models (*left*) routed through technique categories (*middle*) into their AD-VLA adaptation (*right*), grouped by Language Residue level L1–L4.

Recent advances in efficient VLA4AD[1] demonstrate that scale alone does not determine capability: careful language allocation at smaller scales can match or surpass substantially larger baselines.

*This work was supported in part by the NSF CAREER award under grant 2340482 and by NSF award 2414652.
[1]Department of Electrical and Computer Engineering, Northeastern University, Boston, MA, USA. {guo.t, l.su}@northeastern.edu

[1]Vision–Language–Action models for Autonomous Driving, introduced by Jiang et al. (2025b).

Proceedings of the 2026 Conference on Empirical Methods in Natural Language Processing, pages 56933–56947
October 24-29, 2026 ©2026 Association for Computational Linguistics

Table 1: **Comparison of existing surveys on efficient-method technique coverage.** Eff. = efficiency as the main focus. Tech-keys: **Dst**=knowledge distillation, **Pr**=pruning / token reduction, **DS**=dual-system (fast/slow) inference, **Qt**=quantization, **MoE**=mixture-of-experts, **SpA**=sparse / linear attention, **KV**=KV-cache optimization, **LoR**=LoRA / PEFT, **EE**=early-exit, **SpD**=speculative decoding. Status: ✓ = primary coverage, △ = partial / mentioned, ✗ = absent, CLE = Closed loop evaluation.

Survey	Organizing axis	Eff.	Dst	Pr	DS	Qt	MoE	SpA	KV	LoR	EE	SpD	CLE
Yang et al. (2023)	Task taxonomy (QA / planning / generation)	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	△
Cui et al. (2024)	Application domain (perception / planning / HMI)	△	✗	✗	△	△	✗	✗	✗	△	✗	✗	△
Zhou et al. (2024)	VFM role in AD stack	△	△	△	✗	△	✗	✗	✗	△	✗	✗	△
Chen et al. (2024a)	Pipeline placement (modular vs. E2E)	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	✓
Xu et al. (2024a)	On-device LM deployment	△	✓	✓	△	✓	✓	△	△	△	✗	△	✗
Wang et al. (2024)	Small language models taxonomy	△	✓	✓	✗	✓	△	△	△	✓	△	✗	✗
Hu et al. (2025c)	Runtime arch. (E2E vs. dual-system)	△	✗	✗	✓	✗	△	△	✗	✗	✗	✗	✓
Yu et al. (2025)	Efficiency pipeline (model / training / data)	✓	✓	✓	✓	✓	✓	✓	△	✓	△	✓	✗
Sapkota et al. (2025)	VLA design (architecture + training)	△	✗	△	△	△	△	✗	△	△	✗	✗	✗
Sharshar et al. (2025)	Edge deployment (VLM systems)	✓	✓	✓	✗	✓	△	✗	✗	△	✗	✗	✗
Jiang et al. (2025b)	VLA building blocks + evolution	△	△	△	△	△	△	✗	✗	△	✗	✗	✓
Zhu et al. (2025)	Pipeline dichotomy (modular vs. E2E)	△	✗	✗	✗	✗	✗	✗	✗	✗	✗	✗	△
Cai et al. (2025)	Architecture class (Mixture-of-Experts)	△	✗	✗	✗	✗	✓	✗	✗	△	✗	✗	✗
Ours	**Inference-lifecycle language role**	✓	✓	✓	✓	✓	✓	△	✓	✓	✓	✓	✓

Orion-Lite (Gu et al., 2026) distills a 7B teacher into a 0.1B vision-only student that surpasses its teacher on Bench2Drive (Jia et al., 2024) while bypassing the language backbone at inference. DualDriveVLA (Ang et al., 2026) invokes its language module on only about 15% of scenarios and still matches always-on VLM accuracy on NAVSIM v1 (Dauner et al., 2024). Despite unrelated architectures, both *reduce* the language active at inference while preserving its function: reasoning, grounding, and long-tail generalization.

We therefore reframe the design question: not *how small* a model should be, but *when and where language should participate at inference*, since training occurs only once while inference cost is paid at every deployed frame. To organize methods along this axis, we introduce the *Language Residue taxonomy*, which categorizes representative methods by their inference-time use of language: train-time-only supervision (L1), latent non-textual reasoning (L2), conditional invocation (L3), and full per-frame generation (L4). Figure 2 presents the full taxonomy.

The taxonomy spans the major contemporary approaches, which differ in what becomes of language at inference: ① *Knowledge distillation* (Hegde et al., 2025; Gu et al., 2026) relocates its value into the student's weights. ② *Language-supervised pretraining* (Pan et al., 2024; Xu et al., 2024b; Ma et al., 2025) relocates it into learned feature geometry. ③ *Latent world models and flow-matching planners* (Peng et al., 2025; Luo et al., 2026; Xu et al., 2025c; Xing et al., 2025) encode reasoning in continuous representations, dropping the token vocabulary altogether. ④ *Conditional dispatch* (Tian et al., 2024; Ang et al., 2026; Hamdan et al., 2025) invokes language only when a frame demands it: scene-gated routers and asynchronous dual-systems send routine frames down a lightweight reactive path. ⑤ *Within-model efficiency* keeps language active every frame but cuts its cost: mixture-of-experts (MoE) architectures activate only a subset of experts per token (Yang et al., 2026), while quantization, token pruning, and co-designed serving stacks (Sha et al., 2026; Li et al., 2026d) trim latency and memory without altering language's temporal role.

Prior surveys (Table 1) either catalog AD–LLM capabilities (Yang et al., 2023; Cui et al., 2024; Zhou et al., 2024) or organize VLA methods by architecture (Jiang et al., 2025b; Sapkota et al., 2025; Hu et al., 2025c). The closest, Yu et al. (2025) treats efficiency as model compression borrowed from the general LLM (Wang et al., 2024; Xu et al., 2024a) and edge-VLM (Sharshar et al., 2025) literature. It leaves unexamined how much language a deployed system must run at inference, and what determines that allocation in the first place.

We fill that gap by rethinking language as an allocation decision: *retain what language contributes, eliminate what it costs*. We review recent methods (Section 2), annotating each across five deployment axes (Latency↓, Parameters↓, Memory↓, FLOPs↓, Tokens↓), and analyze them on major open- and closed-loop driving benchmarks (Section 4). We further trace how efficient methods from NLP/LLM systems are

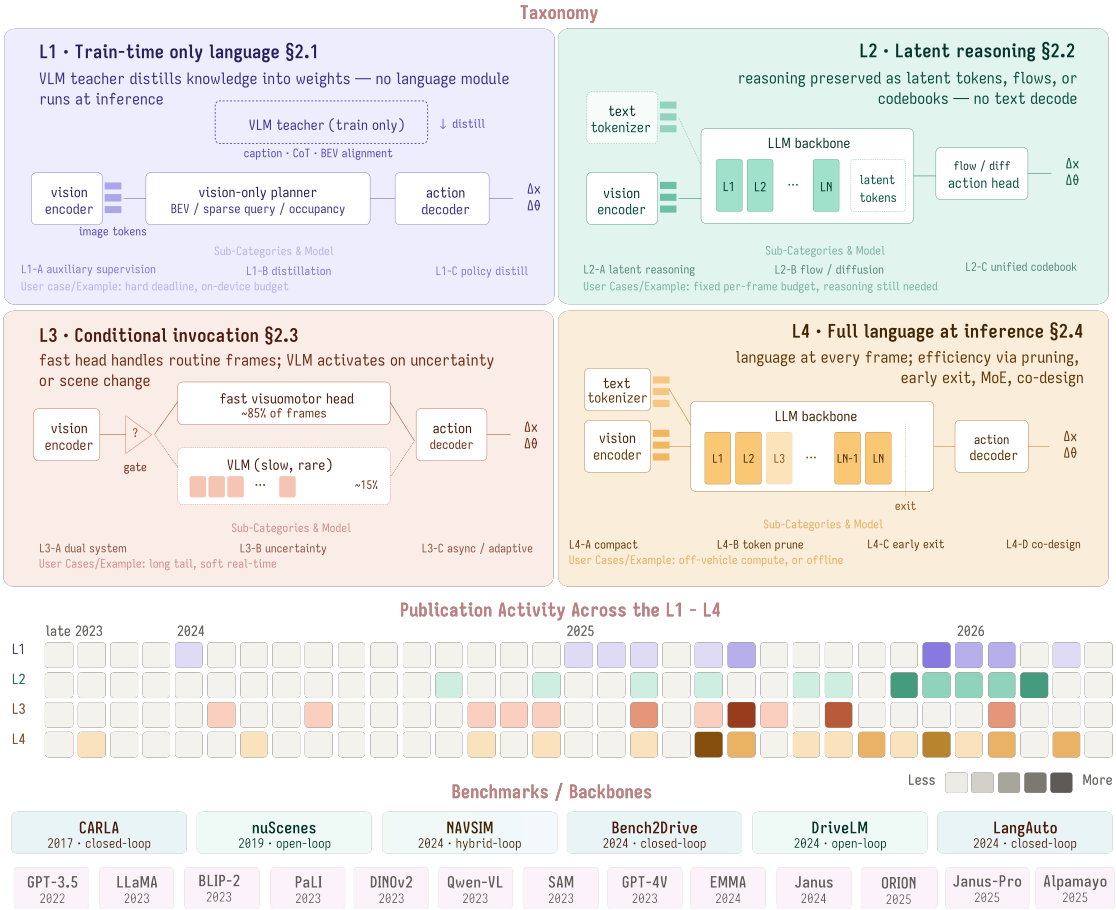

Figure 2: A taxonomy of efficient VLA4AD pipeline. Sensor inputs (multi-camera images, lidar) feed a vision encoder; outputs are consumed by a language backbone that issues planning decisions; an action decoder converts plans into vehicle controls. The Language Residue taxonomy classifies how much language is active at inference, from **L4** (full per-frame autoregressive generation) down to **L1** (no language at runtime). Deployment cost concentrates at the language backbone, motivating the allocation question addressed here.

adapted in AD (Section 3), identifying the constraints and motivations that reshape each under hard-deadline[2] deployment. We close with the open problems that remain (Section 5, Figure 4). We hope this survey, together with our continuously updated repository at `GitHub`, helps researchers and practitioners navigate the rapidly evolving landscape and inspires further advances in efficient VLA design.

2 Language's Role in VLA

2.1 Train-Time-Only Language (L1)

L1 methods remove language from the inference graph entirely: language participates only during training, and the deployed model is a purely visuomotor policy. What varies within L1 is how the language-derived prior reaches the vision-only weights (Table 2).

Auxiliary supervision (L1-A) folds language objectives into the training loss, shaping the gradient without teacher-student distillation. Language thus survives only as a training signal, but that signal is what transfers language-model reasoning into the vision network, buying zero-shot generalization to unseen cities and long-tail maneuvers that a vision-only planner lacks (Pan et al., 2024). *Feature distillation* (L1-B) transfers a VLM's knowledge into a compact planner through a shared scene encoder and surrogate tasks. Because the teacher is discarded after training, the driving path runs language-free yet retains the multimodal LLM's world knowledge in the planner's features, recovering robustness on rare events such as three-point turns and overtaking at no inference cost (Hegde et al., 2025). Above mechanisms trade supervision fidelity against deployment simplicity differently, and none yet dominates across benchmarks.

[2] A fixed control-loop cycle time that cannot be exceeded.

Table 2: **L1 — Train-Time-Only Language Methods.** Language supervises training; no language module executes at inference. • **Benchmarks:** B Bench2Drive (Jia et al., 2024), N nuScenes (Caesar et al., 2020), D DriveLM (Sima et al., 2024). • 🌐 links to an official open-source repository or project page.

#	Method	Backbone / Params	Efficiency	Bench.
• **A** *Auxiliary Supervision — language shapes gradients; inference is purely visuomotor*				
1	VLP (Pan et al., 2024)	UniAD/VAD + text aux.	Latency ↓ Parameters ↓	N
2	VLM-AD (Xu et al., 2024b)	UniAD + GPT-4o	Latency ↓ Parameters ↓	N
3	VLM-E2E (Liu et al., 2025a)	E2E + driver attention	Latency ↓	N
4	OccVLA (Liu et al., 2025b)	VLM + implicit 3D occupancy	FLOPs ↓	N
• **B** *Feature Distillation — VLM teacher transfers representations to a compact student*				
5	RSD (Qin et al., 2025)	E2E + VLM risk, ∼50M	Parameters ↓	B
6	ALN-P3 (Ma et al., 2025)	BEV + LLM-aligned P3	Latency ↓	N
7	VERDI (Feng et al., 2025a)	VLM-embedded reasoning	Latency ↓	N
8	DiMA (Hegde et al., 2025)	VAD + MLLM → VAD-Tiny/Base student	Latency ↓	N D
9	Orion-Lite 🌐 (Gu et al., 2026)	ORION 7B → vision student	Latency ↓ Memory ↓	B

L1 distinguishing property: deploying these models incurs *zero* language-module cost at inference, efficiency gains are fully amortized into weights during training.

2.2 Non-Textual Reasoning (L2)

L2 methods retain reasoning at inference while replacing its textual surface form with continuous latent tokens, carrying intermediate cognition without decoding language. The diversity of L2 designs reflects an unresolved question: which latent substrate best preserves reasoning once explicit text is removed (Table 3).

Continuous latent reasoning (L2-A) replaces text decoding with continuous hidden states to skip generation latency and capture graded uncertainty that discrete tokens cannot (Bai et al., 2026). *Flow- and diffusion-based action heads* (L2-B) replace textual deliberation with a generative distribution over trajectories, recovering the multimodal choice between overtaking and following that unimodal planners lose (Xing et al., 2025). *Unified language–action codebooks* (L2-C) absorb language rather than discard it, interleaving action and text tokens in a single autoregressive sequence. Actions are generated directly, without intermediate rationales, while the LLM's world knowledge and instruction-following remain fully intact.

Across these directions, supervision quality proves more decisive than the choice of substrate. *LaST-VLA* (Luo et al., 2026) isolates the principal failure mode: without explicit supervision, latent tokens degenerate into trivial representations. Its spatio-temporal supervision aligned with physical priors reaches PDMS 91.3 on NAVSIM v1, advancing the state of the art in planning accuracy, while other methods in the tier report the 5–10× speedups over textual CoT that make the level attractive as a whole (e.g., LaRA-VLA's 90% latency reduction (Bai et al., 2026)). LaST-VLA's own paper reports no latency or throughput figure.

2.3 Conditionally Invoked Language (L3)

L3 methods preserve language at inference but invoke it selectively: a lightweight visuomotor module handles routine frames while a heavier language-capable component activates on scene complexity or model uncertainty. The central design freedom at L3 is the *trigger*: the mechanism that decides, per frame, whether language should run at all (Table 4).

Dual-system routing (L3-A) uses a learned gate to classify frames as routine or complex, offering predictable routing overhead but requiring a reliable complexity proxy (Ang et al., 2026). *Uncertainty-triggered invocation* (L3-B) instead reads the fast path's confidence score, making the gate data-driven while inheriting that path's calibration sensitivity (Qian et al., 2024). *Asynchronous and adaptive reasoning* (L3-C) decouples language generation from the control loop entirely, accepting potential staleness in exchange for non-blocking actuation (Luo et al., 2025).

Among trigger designs, the dual-system router currently offers the best-characterized performance boundary: *DualDriveVLA* (Ang et al., 2026) instantiates the canonical dual-system structure: a vision-only ViT paired with a VLM invoked on only ∼15% of scenarios via a learned scorer, reaching 91.00 PDMS on NAVSIM v1 with a 1.9× latency speedup over an always-on VLM baseline. The same paper makes the core trade-off explicit: its *HybridDriveVLA* variant (classified L4 in our taxonomy, since it runs both pathways on every frame with no conditional gating) reaches 92.10 PDMS by running both pathways and selecting the bet-

Table 3: **L2 — Latent Reasoning Methods.** Reasoning executes at inference: no language tokens are decoded. • links to an official open-source repository or project page. • **Benchmarks:** V NAVSIM (Dauner et al., 2024), B Bench2Drive (Jia et al., 2024), N nuScenes (Caesar et al., 2020), I LIBERO (Liu et al., 2023)/SimplerEnv (Li et al., 2024a), R Robotics, A PhysicalAI-AV (NVIDIA, 2025).

#	Method	Backbone / Params	Efficiency	Bench.
• **A** *Latent Reasoning — continuous or structured latent tokens replace textual chain-of-thought*				
1	LCDrive (Tan et al., 2026)	Qwen3-0.5B + world model	Tokens↓ Latency↓	A
2	Reasoning-VLA (Zhang et al., 2025)	VLA + parallel action queries	Latency↓ Tokens↓	N V
3	LaRA-VLA (Bai et al., 2026)	VLA + latent thinking + pred.	Tokens↓ Latency↓	I R
4	LaST-VLA (Luo et al., 2026)	VLM + ST-latent CoT (3D-FM+WM)	Latency↓ Tokens↓	V
5	DynVLA (Shang et al., 2026)	VLA + Dynamics Tokenizer	Latency↓ Tokens↓	V B
6	NetRoller (Xin et al., 2025)	General + specialized interface	Latency↓	N
7	HiST-VLA (Wang et al., 2026c)	LLaVA-v1.5-7B (CLIP ViT-L/14 + Vicuna-7B)	FLOPs↓	V
8	ColaVLA (Peng et al., 2025)	LLaMA-7B + CLR + HPP	Latency↓ Tokens↓	N
9	DrivoR (Kirby et al., 2026)	ViT + register tokens (Valeo)	Latency↓	V
• **B** *Flow / Diffusion Latent Inference — action head replaces autoregressive decoding with one- or few-step generation*				
10	GoalFlow (Xing et al., 2025)	E2E + goal-cond. flow	Latency↓	V
11	WAM-Diff (Xu et al., 2025b)	Llada-V + MoE + GSPO RL	Parameters↓	V
12	ViLaD (Cui et al., 2025)	Vision-language diffusion	Latency↓	N
13	ReCogDrive (Li et al., 2026b)	Cognitive VLM teacher	Latency↓ Parameters↓	V B
14	ReflectDrive (Li et al., 2025a)	Discrete diffusion VLA	Latency↓	V
• **C** *Unified Language–Action Codebooks — shared discrete vocabulary eliminates separate language decoder*				
15	OpenDriveVLA (Zhou et al., 2026)	VLM + discrete actions, 0.5–7B	Latency↓	N

L2 distinguishing property: language structure is *preserved* at inference but in a compressed, non-textual form, enabling semantic reasoning without token-by-token LLM decode cost.

ter output, recovering accuracy but forfeiting the efficiency gain, fixing the frontier that L3 must navigate.

2.4 Full Language at Inference (L4)

L4 methods run the full language pathway on every frame at inference: the language model participates in perception, reasoning, and action generation with no gate and no fallback. Because language runs unconditionally here, the central design freedom at L4 shifts from *whether* to invoke it to *where to reduce its cost*: the compression target within an always-on pathway (Table 5).

Compact-architecture VLAs (L4-A) lower fixed inference cost by choosing smaller backbones from the outset, capping representational capacity in exchange (Fu et al., 2025; Jiang et al., 2025a). *Visual-token pruning* (L4-B) discards uninformative image patches before the LLM stage, cutting attention complexity but bounded by the reliability of the saliency estimate (Xiong et al., 2026). *Dynamic depth and early exit* (L4-C) adapt computation per frame, letting routine inputs leave the network before the final layer while inheriting the calibration sensitivity of the exit criterion (Hu et al., 2025a). *Algorithm–system co-design* (L4-D) co-optimizes across quantization, prefill scheduling, and the hardware–software interface, yielding gains that are hardware-bound rather than portable (Li et al., 2026d).

Among these strategies, compact-architecture results speak most directly to the central L4 question of whether language must be large: *MindDrive* (Fu et al., 2025) matches a 7B-LLM VLA on Bench2Drive using a 0.5B backbone, evidence that at L4 the language model can be scaled down without a proportional accuracy loss. However, evaluation heterogeneity is also highlighted at L4: results rarely share a checkpoint or protocol, leaving it unclear whether efficiency gains on one axis (e.g., visual-token pruning) compound with those on another (e.g., early exit or quantization).

3 NLP/LLM and AD: Migration and Divergence

Core efficiency methods such as quantization, MoE, speculative decoding, PEFT, and distillation stem from LLM prioritizing low token costs and soft latency. Figure 1 illustrates how these methods migrate to AD-VLA architectures. Target-agnostic mechanisms transfer seamlessly, whereas methods dependent on conditions specific to LLMs must be adapted to meet real-time autonomous driving constraints.

Migration landscape. Distillation is repurposed as reasoning supervision without decoding: a language-capable teacher shapes a visuomotor stu-

Table 4: **L3 — Conditionally Invoked Language Methods.** Language module is present but executed only when the driving situation warrants it. • 🌐 links to an official open-source repository or project page. • **Benchmarks:** V NAVSIM (Dauner et al., 2024), B Bench2Drive (Jia et al., 2024), N nuScenes (Caesar et al., 2020), W Waymo (Ettinger et al., 2021), C CARLA (Dosovitskiy et al., 2017), P nuPlan (?).

#	Method	Backbone / Params	Efficiency	Bench.
• **A** *Dual-System Routing — explicit fast (silent) + slow (language) paths; VLM invoked per frame subset*				
1	LeapAD 🌐 (Mei et al., 2024)	VLM (Qwen-VL-7B) dual-process + memory	Latency↓	C
2	DualDriveVLA (Ang et al., 2026)	ViT + VLM + scorer	Latency↓	V
• **B** *Uncertainty-Triggered Invocation — language module activates when fast path signals insufficient confidence*				
3	FASIONAD (Qian et al., 2024)	ViT + VLM + adaptive feedback	Latency↓	N
4	FASIONAD++ (Qian et al., 2025)	VAD/GenAD + VLM + uncertainty gate	Latency↓	N B
• **C** *Asynchronous & Adaptive Reasoning — language tokens generated at variable length or out-of-phase with control*				
5	DriveVLM-Dual (Tian et al., 2024)	Qwen-VL 9.6B + VAD planner	Latency↓	N
6	SOLVE / SOLVE-VLM (Chen et al., 2025)	VLM + E2E network synergy	Latency↓	N
7	ETA 🌐 (Hamdan et al., 2025)	LLaVA-Next ∼7B + fast ctrl	Latency↓	B
8	AutoVLA 🌐 (Zhou et al., 2025)	Qwen2.5-VL-3B + GRPO RFT	Tokens↓ Latency↓	W P N C
9	AdaThinkDrive (Luo et al., 2025)	VLM + adaptive thinking RL	Tokens↓	V
10	ReAL-AD (Lu et al., 2025)	VLM + human-like multi-stage	Latency↓	N B

L3 distinguishing property: the language module is a run-time component, but most frames never invoke it. Language cost scales with workload complexity, not with wall-clock time.

dent that incurs zero language cost at runtime (Gu et al., 2026; Hegde et al., 2025). MoE routing is driven by scene context rather than token identity, exploiting the long-tail distribution of driving situations (Yang et al., 2026). Speculative decoding inspires fast–slow pipelines where the "draft" is a short-horizon plan or meta-action verified by a slower VLM rather than a token prefix, giving a fixed dual-frequency architecture in place of the LLM's elastic draft length (Tian et al., 2024). Early exit transfers, but its trigger flips from token-confidence to physical feasibility: a trajectory within prior bounds is the new high-probability token, and the exit criterion is grounded in safety margin rather than output entropy (Hu et al., 2025a). Quantization and PEFT/LoRA transfer most directly (Frantar et al., 2023; Lin et al., 2024; Hu et al., 2022); visual-token and KV-cache pruning shift the efficiency bottleneck from decode (LLM) to prefill (AD-VLA) (Sha et al., 2026; Xiong et al., 2026). Serving-layer optimizations (e.g., continuous batching, PagedAttention (Kwon et al., 2023)) do not transfer: they presuppose batchable independent requests, structurally absent from AD's single-stream deployment.

Most aggressive reshapings at L1 and L2. L1 and L2 represent AD pushing past its LLM teachers. Distillation collapses into a regime where language is fully erased from the inference graph rather than merely compressed (Pan et al., 2024; Ma et al., 2025; Gu et al., 2026): the student inherits no chain-of-thought at runtime, not just a smaller one. Latent-reasoning work treats register tokens and meta-action symbols as first-class outputs (Fu et al., 2025; Peng et al., 2025; Kirby et al., 2026; Jiang et al., 2024b) in a way mainstream LLM systems have only just begun to explore. The unifying thread: computation gates are *task-structural and externally grounded*, not model-internal.

Four patterns are either AD-specific or substantially reshaped by its constraints: *(i)* asynchronous fast–slow inference clocked at distinct fixed frequencies (Tian et al., 2024; Hamdan et al., 2025); *(ii)* meta-action shortcuts collapsing a planning segment into one discrete token (Jiang et al., 2024b); *(iii)* train-time-only language supervision fully removed from the inference graph (Pan et al., 2024; Ma et al., 2025); *(iv)* scene-gated computation conditioned on external driving context rather than model confidence (Yang et al., 2026).

Reverse influence: a task-structural view of when language helps. Selective language invocation has been studied extensively in LLM systems: model cascades (Chen et al., 2024b), adaptive and budget-controlled CoT (Han et al., 2025), confidence-based early exit (Schuster et al., 2022), speculative decoding (Leviathan et al., 2023; Li et al., 2024b), sparse MoE routing (Shazeer et al., 2017; Jiang et al., 2024a), and retrieval/tool gating in agent systems.

What distinguishes AD is not *whether* language is gated, but *what the gate conditions on*. LLM-system gating is predominantly driven by model-internal signals (e.g. token confidence, output en-

Table 5: **L4 — Full Language at Inference.** Language generated every frame; efficiency via smaller models or sparser visual inputs. • **Benchmarks:** V NAVSIM (Dauner et al., 2024), B Bench2Drive (Jia et al., 2024), N nuScenes (Caesar et al., 2020), D DriveLM (Sima et al., 2024), C CARLA (Dosovitskiy et al., 2017), W Waymo (Ettinger et al., 2021), A PhysicalAI-AV (NVIDIA, 2025). • 🌐 links to an official open-source repository or project page.

#	Method	Backbone / Params	Efficiency	Bench.
• **A** *Compact-Architecture VLAs — small or lightweight backbone reduces parameter / memory cost*				
1	LMDrive 🌐 (Shao et al., 2024)	LLaMA-7B + custom vision encoder	Parameters↓	C
2	EM-VLM4AD 🌐 (Gopalkrishnan et al., 2024)	Lightweight multimodal VLM	Memory↓ FLOPs↓	D
3	FastDrive (Jiang et al., 2025a)	0.9B VLM, structured output	Parameters↓ Latency↓	N
4	DSDrive (Liu et al., 2025c)	Compact LLM + dual-head	Parameters↓ Latency↓	C
5	Drive-KD (Lian et al., 2026)	VLM (InternVL3/Qwen2.5-VL) → smaller VLM	Latency↓ Parameters↓	N
6	MindDrive (Fu et al., 2025)	Qwen2-0.5B + dual-LoRA (Decision/Action experts)	Parameters↓ Latency↓	B
• **B** *Visual-Token Pruning — reduce LLM input length by selecting informative image tokens*				
7	FastDriveVLA (Cao et al., 2026)	Plug-in ReconPruner (MAE)	FLOPs↓ Latency↓	N
8	Prune2Drive 🌐 (Xiong et al., 2026)	Multi-view diversity pruner	FLOPs↓ Memory↓	D
9	ETA-VLA (Wang et al., 2026d)	Text-guided token scoring	FLOPs↓	V
10	ST-Prune (Sha et al., 2026)	Spatio-temporal pruner	Tokens↓ Latency↓	D
• **C** *Dynamic Depth & Early Exit — adapt computation depth per sample; exit before final layer when confident*				
11	DeeAD (Hu et al., 2025a)	ORION + action-guided exit	Latency↓ FLOPs↓	B
12	Nav-EE (Hu et al., 2025b)	VLM + navigation-guided exit	Latency↓	W
• **D** *Algorithm–System Co-Design — joint optimization of model and serving stack; quantization and MoE routing included*				
13	DriveMoE (Yang et al., 2026)	MoE VLA (Vision router + 1-shared/6-expert Action MoE)	Tokens↓	B N
14	FlashDriveVLA 🌐 (Li et al., 2026d)	Alpamayo 1/1.5-10B, full pipeline	Latency↓ Memory↓	A
15	LinkVLA (Wang et al., 2026b)	InternVL2-1B (InternViT-300M + Qwen2-0.5B) + discrete codebook	Latency↓	C

L4 distinguishing property: full per-frame language generation is preserved; efficiency gains come from *within* the language module rather than from reducing when language is used.

tropy, learned reasoning budgets) under soft latency targets. AD's hard-deadline, single-stream setting forces gates conditioned on *external task structure*: scene complexity, agreement with a planning prior, dynamics-conditioned thresholds, and asynchronous clocks tied to physical control rates. We suggest this task-structural view of conditional language computation is the transferable insight in the AD→LLM-systems direction, particularly for latency-critical applications where gating signals can be grounded in the environment or task contract rather than in the language model's own uncertainty. The pattern, taken together, is that AD adopts LLM efficiency techniques where they accelerate computation but invents its own where the mechanism itself imposes language load, and the Language Residue taxonomy is what makes the asymmetry legible.

Figure 3: Main Benchmark Coverage by Language Residue Taxonomy. Each cell reports the number of methods at that level with published results on the named benchmark. A method may appear in multiple columns. Only widely used benchmarks are plotted.

4 Benchmarks and Evaluation

4.1 Open-Loop Benchmarks

nuScenes (Caesar et al., 2020) measures trajectory-prediction accuracy over 1,000 real-world driving scenes (1.4M camera frames), using L2 displacement error (m) and collision rate (%) as primary metrics. It is the most-reported benchmark in this survey (Figure 3), though its use of ego-status as a model input has drawn sustained criticism for leaking ground-truth velocity information.

NAVSIM (Dauner et al., 2024) scores open-loop trajectories against a non-reactive closed-loop oracle, collapsing multiple driving competencies into a single Predictive Driver Model Score (PDMS; v2: EPDMS). It is the primary substrate for L2 methods, and supplanted nuScenes as the standard

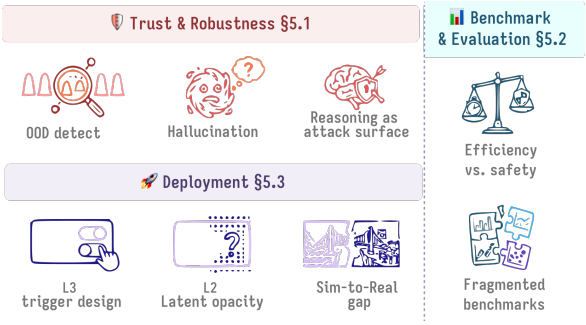

Figure 4: Open problems in efficient VLA4AD. Three tiers of gaps: trust and robustness, benchmark and evaluation, and deployment.

open-loop evaluation platform from 2025 onward.

4.2 Closed-Loop Benchmarks

Bench2Drive (Jia et al., 2024) tests interactive closed-loop driving across 44 routes and 220 CARLA scenarios, measuring Driving Score (DS) and Success Rate (SR) across multi-ability test conditions.

CARLA Leaderboard 2.0 extends the CARLA simulator (Dosovitskiy et al., 2017) to long routes (∼10 km) with over 20 hazard types, reporting DS, route completion, and infraction counts.

DriveLM (Sima et al., 2024) evaluates language-grounded reasoning through a Graph Visual Question Answering (GVQA) protocol that chains perception, prediction, and planning over nuScenes and CARLA scenes (∼440K QA pairs). It reports a composite Final Score that aggregates accuracy on multiple-choice questions, a GPT-judged score for planning answers, language-generation metrics (BLEU, ROUGE-L, CIDEr) for perception, and an object-matching score for prediction.

Waymo Open Motion Dataset (WOMD) (Ettinger et al., 2021) benchmarks long-horizon trajectory prediction and planning over 103K real-world scenes, reporting minimum average displacement error (minADE) and minimum final displacement error (minFDE).

5 Open Problems

5.1 Trustworthiness and Robustness

☆ *Out-of-distribution (OOD) is underexplored.* Language models are typically kept on the inference path specifically to navigate the long tail (Wang et al., 2025b; Fang et al., 2025). However, as Chi et al. (2025) note, driving VLAs still fail in unstructured corner cases due to a scarcity of targeted benchmarks. This creates a paradox: the component tasked with absorbing distribution shifts sits precisely where alignment data is thinnest. Conditional invocation systems (Hamdan et al., 2025; Zhou et al., 2025; Luo et al., 2025) turn this gap into a functional risk. By dynamically gating the language path, they implicitly guess whether a scene falls outside the fast-path distribution, without validating those choices against any shift criterion. The irony is that the required machinery already exists. Formulations for runtime OOD detection (Guo et al., 2026) and independent behavior benchmarks (Distelzweig et al., 2026) are both available, yet entirely absent from current evaluations. ☆

Hallucination is introduced by design. Language supervision rewards consistency with language priors rather than factual observation. Thus, unpenalized misreadings easily poison downstream planning. Fan et al. (2024) confirm this stems from the objective itself, reporting massive F1 gains when cross-checking is enforced. Crucially, this failure requires no adversarial trigger, making it a common operational risk that security-focused research largely ignores. Mitigation strategies target different points of intervention: reinforcement learning evaluates the reasoning trace by driving outcomes (Li et al., 2025b; Yuan et al., 2025); auxiliary objectives anchor semantic outputs to geometric predictions (Liu et al., 2025b; Ma et al., 2025); and runtime supervision filters narrations before they reach the controller (Li et al., 2026c). Because these methods lack a unified hallucination metric, their comparative effectiveness remains unknown.

☆ *Attacks reach action through reasoning.* Driving VLMs face severe vulnerabilities across both white-box (Zhang et al., 2026) and black-box (Wang et al., 2025a) regimes, where corrupting the reasoning process directly corrupts the trajectory. Conditional invocation introduces a secondary target: because the gate defines a learnable mapping from an observable condition to a behavioral change, adversaries can trigger malicious behavior through common physical objects (Ni et al., 2024) or achieve 90% attack success at a mere 10% poisoning ratio (Wang et al., 2026a). Countermeasures remain partial and lack level-agnostic guarantees. Pre-deployment evaluation tools like scenario generation and fault injection (Xu et al., 2022; Pasandideh and Rettberg, 2026) target the general driving stack rather than the language path specifically. At runtime, supervision over emitted narrations (Li et al., 2026c) and distribution-shift detection (Guo et al.,

2026) serve to detect and degrade rather than prevent attacks, with the former requiring an explicitly exposed language channel. On the model side, the sole VLA attack-defense framework is limited to manipulation policies (Xu et al., 2025a), and broader defense mechanisms (Li et al., 2026a) are completely unevaluated for driving.

5.2 Benchmark and Evaluation

☆ *Efficiency–safety frontier is unquantified.* Most work reports latency or FLOPs reductions without measuring whether reduced compute raises infraction rates under edge-case scenarios. This gap is confirmed by Xu et al. (2022), who find that autonomous driving algorithms exhibit a direct performance trade-off between benign and safety-critical scenarios, a trade-off that is rarely characterized jointly in efficiency-focused papers. Moreover, Pasandideh and Rettberg (2026) demonstrate that under sensor-degradation faults such as fog, RMSE can increase by up to 99% relative to clean-data baselines, yet such stress conditions are almost never included in compute-efficiency evaluations.

☆ *Benchmark fragmentation prevents cross-level comparison.* Only a small fraction of surveyed methods report on both a trajectory-planning benchmark and a language/reasoning benchmark, leaving open whether efficiency gains in planning come at the cost of reasoning capability. This fragmentation is structural: trajectory benchmarks (nuPlan, NAVSIM, Bench2Drive) and language-QA or scene-understanding benchmarks (DriveLM, DriveCombo, VLM safety suites) are evaluated by disjoint communities with incompatible metrics. Distelzweig et al. (2026) argue that the most widely-used planning benchmark, nuPlan Val14, "can be largely solved by lane-following combined with simple collision-checking," meaning high scores do not imply genuine reasoning capability. Conversely, Ma et al. (2026) show that state-of-the-art multimodal models degrade sharply on traffic-rule conflict tasks (L5 of their five-level cognitive ladder) despite strong trajectory-level numbers. No existing work bridges these two evaluation axes in a single controlled study.

5.3 Deployment and Future Gap

☆ *The L3 trigger design problem has no principled solution.* Adaptive-compute systems must decide *when* to engage deliberative reasoning rather than reactive control, yet no criterion distinguishes deterministic clocks, entropy thresholds, and learned gates, and the safety implications of trigger choice remain unexplored. AdaThinkDrive (Luo et al., 2025) recently demonstrated that a VLA framework with a "Fast answering / Slow thinking" mechanism can reduce inference time by 14% on NAVSIM, but the trigger design is evaluated purely for efficiency; the safety consequences of misclassifying a scene as "easy" are not examined.

☆ *L2 latent reasoning resists inspection.* Efficiency gains from compressing cognition into latent representations come at the cost of interpretability. A comprehensive survey of world models for autonomous driving (Feng et al., 2025b) identifies "opaque latent policies" as one of three persistent open issues alongside generalization and formal safety guarantees.

☆ *Real-world deployment evidence is sparse.* The large majority of surveyed methods are validated exclusively in simulation, leaving the sim-to-real transfer of both efficiency and safety properties unknown. Hu et al. (2024) identify differences in lighting, textures, vehicle dynamics, and agent behavior as persistent sources of the reality gap. Arango et al. (2026) find that discrepancies in ego-vehicle dynamics, sensor behavior, and processing times introduce a performance gap that survives domain-randomization and digital-twin transfer.

6 Conclusion

We organized the emerging design space of efficient VLA for autonomous driving around a single question: how much language computation should survive to inference? The Language Residue taxonomy (L1–L4) provides a principled answer space, classifying representative methods by their inference-time language involvement and annotating each across five deployment axes (Latency, Parameters, Memory, FLOPs, Tokens). Analysis of the NLP/LLM–AD efficiency genealogy reveals a selective migration: techniques that accelerate computation transfer with adaptation, while techniques whose mechanism is language-load are replaced by AD-native designs grounded in task structure. Cross-level benchmark analysis exposes fragmentation as a primary barrier to fair comparison and efficient progress. We hope the taxonomy and the continuously updated repository at `GitHub` serve as shared vocabulary for the community and accelerate progress toward efficient, safe, and interpretable autonomous driving.

Limitations

Several limitations are worth noting. Coverage is weighted toward methods in which a language module sits on the inference path, since the Language Residue levels are defined by that module's presence. Earlier language-conditioned and modular driving systems are cited for context but not assigned levels, as they do not expose a comparable inference-time language budget. Concurrent work not yet publicly available is also outside our coverage. Benchmark analysis is constrained by what authors self-report: efficiency metrics (FLOPs, latency, memory) and driving performance metrics rarely appear in the same paper, limiting our ability to characterize the efficiency–accuracy Pareto frontier. The large majority of surveyed methods are evaluated in simulation; real-world deployment results are available for only a handful of systems (Tian et al., 2024), leaving the simulation-to-real transfer of efficiency claims uncharacterized. Reported gains are platform-bound: latency and memory hold only for the hardware measured, while parameter counts, FLOPs, and token counts transfer across platforms. Few methods report timings on automotive-grade hardware, so speedups should be read as relative. The taxonomy assigns each method a single primary Language Residue level, which may oversimplify architectures that combine, for example, L1-style pretraining with L3-style conditional dispatch at inference.

References

Sining Ang, Yuguang Yang, Chenxu Dang, Canyu Chen, Cheng Chi, Haiyan Liu, Xuanyao Mao, Jason Bao, Xuliang, Bingchuan Sun, and Yan Wang. 2026. From representational complementarity to dual systems: Synergizing VLM and vision-only backbones for end-to-end driving. *Preprint*, arXiv:2602.10719.

J. Felipe Arango, Rodrigo Gutiérrez-Moreno, Pedro A. Revenga, Ángel Llamazares, Elena López-Guillén, and Luis M. Bergasa. 2026. Closing sim2real gaps: A versatile development and validation platform for autonomous driving stacks. *Sensors*, 26(4):1338.

Shuanghao Bai, Jing Lyu, Wanqi Zhou, Zhe Li, Dakai Wang, Lei Xing, Xiaoguang Zhao, Pengwei Wang, Zhongyuan Wang, Cheng Chi, Badong Chen, and Shanghang Zhang. 2026. Latent reasoning VLA: Latent thinking and prediction for vision-language-action models. In *Proceedings of the 43rd International Conference on Machine Learning*.

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

Weilin Cai, Juyong Jiang, Fan Wang, Jing Tang, Sunghun Kim, and Jiayi Huang. 2025. A survey on mixture of experts in large language models. *IEEE Transactions on Knowledge and Data Engineering*.

Jiajun Cao, Qizhe Zhang, Peidong Jia, Xuhui Zhao, Bo Lan, Xiaoan Zhang, Zhuo Li, Xiaobao Wei, Sixiang Chen, Liyun Li, Xianming Liu, Ming Lu, Yang Wang, and Shanghang Zhang. 2026. FastDriveVLA: Efficient end-to-end driving via plug-and-play reconstruction-based token pruning. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

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

Lingjiao Chen, Matei Zaharia, and James Zou. 2024b. FrugalGPT: How to use large language models while reducing cost and improving performance. *Transactions on Machine Learning Research*.

Xuesong Chen, Linjiang Huang, Tao Ma, Rongyao Fang, Shaoshuai Shi, and Hongsheng Li. 2025. SOLVE: Synergy of language-vision and end-to-end networks for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Haohan Chi, Huan-ang Gao, Ziming Liu, Jianing Liu, Chenyu Liu, Jinwei Li, Kaisen Yang, Yangcheng Yu, Zeda Wang, Wenyi Li, Leichen Wang, Xingtao Hu, Hao Sun, Hang Zhao, and Hao Zhao. 2025. Impromptu VLA: Open weights and open data for driving vision-language-action models. *Preprint*, arXiv:2505.23757.

Can Cui, Yunsheng Ma, Sung-Yeon Park, Zichong Yang, Yupeng Zhou, Peiran Liu, Juanwu Lu, Juntong Peng, Jiaru Zhang, Ruqi Zhang, Lingxi Li, Yaobin Chen, Jitesh H. Panchal, Amr Abdelraouf, Rohit Gupta, Kyungtae Han, and Ziran Wang. 2024. LLM4AD: Large language models for autonomous driving. *Preprint*, arXiv:2410.15281.

Can Cui, Yupeng Zhou, Juntong Peng, Sung-Yeon Park, Zichong Yang, Prashanth Sankaranarayanan, Jiaru Zhang, Ruqi Zhang, and Ziran Wang. 2025. ViLaD: A large vision language diffusion framework for end-to-end autonomous driving. *Preprint*, arXiv:2508.12603.

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

Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track.

Aron Distelzweig, Faris Janjoš, Andreas Look, Anna Rottenhäusler, Daniel Jost, Oliver Scheel, Raghu Rajan, Daphne Cornelisse, Eugene Vinitsky, and Joschka Boedecker. 2026. Beyond self-play and scale: A behavior benchmark for generalization in autonomous driving. *arXiv preprint arXiv:2605.10034*.

Alexey Dosovitskiy, German Ros, Felipe Codevilla, Antonio Lopez, and Vladlen Koltun. 2017. CARLA: An open urban driving simulator. In *Proceedings of the 1st Annual Conference on Robot Learning*.

Scott Ettinger, Shuyang Cheng, Benjamin Caine, Chenxi Liu, Hang Zhao, Sabeek Pradhan, Yuning Chai, Ben Sapp, Charles R. Qi, Yin Zhou, Zoey Yang, Aurélien Chouard, Pei Sun, Jiquan Ngiam, Vijay Vasudevan, Alexander McCauley, Jonathon Shlens, and Dragomir Anguelov. 2021. Large scale interactive motion forecasting for autonomous driving: The Waymo Open Motion Dataset. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Jiaqi Fan, Jianhua Wu, Hongqing Chu, Quanbo Ge, and Bingzhao Gao. 2024. Hallucination elimination and semantic enhancement framework for vision-language models in traffic scenarios. *arXiv preprint arXiv:2412.07518*.

Shiyu Fang, Yiming Cui, Haoyang Liang, Chen Lv, Peng Hang, and Jian Sun. 2025. CoReVLA: A dual-stage end-to-end autonomous driving framework for long-tail scenarios via collect-and-refine. *Preprint*, arXiv:2509.15968.

Bowen Feng, Zhiting Mei, Julian Ost, Filippo Ghilotti, Baiang Li, Roger Girgis, Anirudha Majumdar, and Felix Heide. 2025a. VERDI: VLM-embedded reasoning for autonomous driving. *Preprint*, arXiv:2505.15925.

Tuo Feng, Wenguan Wang, and Yi Yang. 2025b. A survey of world models for autonomous driving. *arXiv preprint arXiv:2501.11260*.

Elias Frantar, Saleh Ashkboos, Torsten Hoefler, and Dan Alistarh. 2023. GPTQ: Accurate post-training quantization for generative pre-trained transformers. In *Proceedings of the Eleventh International Conference on Learning Representations*.

Haoyu Fu, Diankun Zhang, Zongchuang Zhao, Jianfeng Cui, Hongwei Xie, Bing Wang, Guang Chen, Dingkang Liang, and Xiang Bai. 2025. MindDrive: A vision-language-action model for autonomous driving via online reinforcement learning. *Preprint*, arXiv:2512.13636.

Akshay Gopalkrishnan, Ross Greer, and Mohan Trivedi. 2024. Multi-frame, lightweight and efficient vision-language models for question answering in autonomous driving. In *CVPR 2024 Workshop on Vision and Language for Autonomous Driving and Robotics (VLADR)*.

Jing Gu, Niccolo Cavagnero, and Gijs Dubbelman. 2026. Orion-lite: Distilling LLM reasoning into efficient vision-only driving models. *Preprint*, arXiv:2604.08266.

Tongfei Guo, Taposh Banerjee, Rui Liu, and Lili Su. 2026. Building real-time awareness of out-of-distribution in trajectory prediction for autonomous vehicles. *ACM Transactions on Cyber-Physical Systems*. Just Accepted.

Shadi Hamdan, Chonghao Sima, Zetong Yang, Hongyang Li, and Fatma Güney. 2025. ETA: Efficiency through thinking ahead, a dual approach to self-driving with large models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Tingxu Han, Zhenting Wang, Chunrong Fang, Shiyu Zhao, Shiqing Ma, and Zhenyu Chen. 2025. Token-budget-aware LLM reasoning. In *Findings of the Association for Computational Linguistics: ACL 2025*, pages 24842–24855.

Deepti Hegde, Rajeev Yasarla, Hong Cai, Shizhong Han, Apratim Bhattacharyya, Shweta Mahajan, Litian Liu, Risheek Garrepalli, Vishal M. Patel, and Fatih Porikli. 2025. Distilling multi-modal large language models for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Edward J. Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *Proceedings of the Tenth International Conference on Learning Representations*.

Haibo Hu, Lianming Huang, Nan Guan, and Chun Jason Xue. 2025a. DeeAD: Dynamic early exit of vision-language action for efficient autonomous driving. *Preprint*, arXiv:2511.20720.

Haibo Hu, Lianming Huang, Xinyu Wang, Yufei Cui, Shangyu Wu, Nan Guan, and Chun Jason Xue. 2025b. Nav-EE: Navigation-guided early exiting for efficient vision-language models in autonomous driving. *Preprint*, arXiv:2510.01795.

Tianshuai Hu, Xiaolu Liu, Song Wang, Yiyao Zhu, Ao Liang, Lingdong Kong, Guoyang Zhao, Zeying Gong, Jun Cen, Zhiyu Huang, Xiaoshuai Hao, Linfeng Li, Hang Song, Xiangtai Li, Jun Ma, Shaojie Shen, Jianke Zhu, Dacheng Tao, Ziwei Liu, and Junwei Liang. 2025c. Vision-language-action models for autonomous driving: Past, present, and future. *Preprint*, arXiv:2512.16760.

Xuemin Hu, Shen Li, Tingyu Huang, Bo Tang, Rouxing Huai, and Long Chen. 2024. How simulation helps autonomous driving: A survey of sim2real, digital twins, and parallel intelligence. *IEEE Transactions on Intelligent Vehicles*, 9:593–612.

Zilin Huang, Zihao Sheng, Zhengyang Wan, Yansong Qu, Junwei You, Sicong Jiang, and Sikai Chen. 2026. DriveVLM-RL: Neuroscience-inspired reinforcement learning with vision-language models for safe and deployable autonomous driving. *arXiv preprint arXiv:2603.18315*.

Xiaosong Jia, Zhenjie Yang, Qifeng Li, Zhiyuan Zhang, and Junchi Yan. 2024. Bench2drive: Towards multi-ability benchmarking of closed-loop end-to-end autonomous driving. In *Advances in Neural Information Processing Systems*.

Albert Q. Jiang, Alexandre Sablayrolles, Antoine Roux, Arthur Mensch, Blanche Savary, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Emma Bou Hanna, Florian Bressand, Gianna Lengyel, Guillaume Bour, Guillaume Lample, Lélio Renard Lavaud, Lucile Saulnier, Marie-Anne Lachaux, Pierre Stock, Sandeep Subramanian, Sophia Yang, and 7 others. 2024a. Mixtral of experts. *Preprint*, arXiv:2401.04088.

Bo Jiang, Shaoyu Chen, Bencheng Liao, Xingyu Zhang, Wei Yin, Qian Zhang, Chang Huang, Wenyu Liu, and Xinggang Wang. 2024b. Senna: Bridging large vision-language models and end-to-end autonomous driving. *Preprint*, arXiv:2410.22313.

Hao Jiang, Chuan Hu, Yukang Shi, Yuan He, Ke Wang, Xi Zhang, and Zhipeng Zhang. 2025a. Structured labeling enables faster vision-language models for end-to-end autonomous driving. *Preprint*, arXiv:2506.05442.

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. 2025b. A survey on vision-language-action models for autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision Workshops*.

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

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient memory management for large language model serving with PagedAttention. In *Proceedings of the 29th Symposium on Operating Systems Principles*.

Yaniv Leviathan, Matan Kalman, and Yossi Matias. 2023. Fast inference from transformers via speculative decoding. In *Proceedings of the 40th International Conference on Machine Learning*.

Pengxiang Li, Yinan Zheng, Yue Wang, Huimin Wang, Hang Zhao, Jingjing Liu, Xianyuan Zhan, Kun Zhan, and Xianpeng Lang. 2025a. Discrete diffusion for reflective vision-language-action models in autonomous driving. *Preprint*, arXiv:2509.20109.

Qi Li, Bo Yin, Weiqi Huang, Ruhao Liu, Bojun Zou, Runpeng Yu, Jingwen Ye, Weihao Yu, and Xinchao Wang. 2026a. Vision-language-action safety: Threats, challenges, evaluations, and mechanisms. *arXiv preprint arXiv:2604.23775*.

Xuanlin Li, Kyle Hsu, Jiayuan Gu, Karl Pertsch, Oier Mees, Homer Rich Walke, Chuyuan Fu, Ishikaa Lunawat, Isabel Sieh, Sean Kirmani, Sergey Levine, Jiajun Wu, Chelsea Finn, Hao Su, Quan Vuong, and Ted Xiao. 2024a. Evaluating real-world robot manipulation policies in simulation. In *Conference on Robot Learning (CoRL)*.

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

Yue Li, Meng Tian, Dechang Zhu, Jiangtong Zhu, Zhenyu Lin, Zhiwei Xiong, and Xinhai Zhao. 2025b. Drive-R1: Bridging reasoning and planning in VLMs for autonomous driving with reinforcement learning. *Preprint*, arXiv:2506.18234.

Yuhui Li, Fangyun Wei, Chao Zhang, and Hongyang Zhang. 2024b. EAGLE: Speculative sampling requires rethinking feature uncertainty. In *Proceedings of the 41st International Conference on Machine Learning (ICML)*, volume 235, pages 28935–28948.

Yun Li, Yidu Zhang, Simon Thompson, Ehsan Javanmardi, and Manabu Tsukada. 2026c. Causal scene narration with runtime safety supervision for vision-language-action driving. *arXiv preprint arXiv:2604.01723*.

Zekai Li, Yihao Liang, Hongfei Zhang, Jian Chen, and Zhijian Liu. 2026d. FlashDrive: Flash vision-language-action inference for autonomous driving. Workshop on Efficient Systems for Foundation Models at ICLR 2026. Available at `https://openreview.net/forum?id=kuZrNI5oZM`. No arXiv preprint.

Weitong Lian, Zecong Tang, Haoran Li, Tianjian Gao, Yifei Wang, Zixu Wang, Lingyi Meng, Tengju Ru, Zhejun Cui, Yichen Zhu, Hangshuo Cao, Qi Kang, Tianxing Chen, Kaixuan Wang, and Yu Zhang. 2026. Drive-KD: Multi-teacher distillation for VLMs in autonomous driving. *Preprint*, arXiv:2601.21288.

Ji Lin, Jiaming Tang, Haotian Tang, Shang Yang, Wei-Ming Chen, Wei-Chen Wang, Guangxuan Xiao, Xingyu Dang, Chuang Gan, and Song Han. 2024. AWQ: Activation-aware weight quantization for LLM compression and acceleration. In *Proceedings of Machine Learning and Systems*.

Bo Liu, Yifeng Zhu, Chongkai Gao, Yihao Feng, Qiang Liu, Yuke Zhu, and Peter Stone. 2023. LIBERO: Benchmarking knowledge transfer for lifelong robot learning. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*.

Pei Liu, Haipeng Liu, Haichao Liu, Xin Liu, Jinxin Ni, and Jun Ma. 2025a. VLM-E2E: Enhancing end-to-end autonomous driving with multimodal driver attention fusion. *Preprint*, arXiv:2502.18042.

Ruixun Liu, Lingyu Kong, Derun Li, and Hang Zhao. 2025b. OccVLA: Vision-language-action model with implicit 3D occupancy supervision. *Preprint*, arXiv:2509.05578.

Wenru Liu, Pei Liu, and Jun Ma. 2025c. DSDrive: Distilling large language model for lightweight end-to-end autonomous driving with unified reasoning and planning. *Preprint*, arXiv:2505.05360.

Yuhang Lu, Jiadong Tu, Yuexin Ma, and Xinge Zhu. 2025. ReAL-AD: Towards human-like reasoning in end-to-end autonomous driving. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*.

Yuechen Luo, Fang Li, Shaoqing Xu, Yang Ji, Zehan Zhang, Bing Wang, Yuannan Shen, Jianwei Cui, Long Chen, Guang Chen, Hangjun Ye, Zhi-Xin Yang, and Fuxi Wen. 2026. LaST-VLA: Thinking in latent spatio-temporal space for vision-language-action in autonomous driving. *Preprint*, arXiv:2603.01928.

Yuechen Luo, Fang Li, Shaoqing Xu, Zhiyi Lai, Lei Yang, Qimao Chen, Ziang Luo, Zixun Xie, Shengyin Jiang, Jiaxin Liu, Long Chen, Bing Wang, and Zhixin Yang. 2025. AdaThinkDrive: Adaptive thinking via reinforcement learning for autonomous driving. *Preprint*, arXiv:2509.13769.

Enhui Ma, Jiahuan Zhang, Guantian Zheng, Tao Tang, Shengbo Eben Li, Yuhang Lu, Xia Zhou, Xueyang Zhang, Yifei Zhan, Kun Zhan, Zhihui Hao, Xianpeng Lang, and Kaicheng Yu. 2026. DriveCombo: Benchmarking compositional traffic rule reasoning in autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yunsheng Ma, Burhaneddin Yaman, Xin Ye, Mahmut Yurt, Jingru Luo, Abhirup Mallik, Ziran Wang, and Liu Ren. 2025. ALN-P3: Unified language alignment for perception, prediction, and planning in autonomous driving. *Preprint*, arXiv:2505.15158.

Jianbiao Mei, Yukai Ma, Xuemeng Yang, Licheng Wen, Xinyu Cai, Xin Li, Daocheng Fu, Bo Zhang, Pinlong Cai, Min Dou, Botian Shi, Liang He, Yong Liu, and Yu Qiao. 2024. Continuously learning, adapting, and improving: A dual-process approach to autonomous driving. In *Advances in Neural Information Processing Systems*.

Zhenyang Ni, Rui Ye, Yuxi Wei, Zhen Xiang, Yanfeng Wang, and Siheng Chen. 2024. Physical backdoor attack can jeopardize driving with vision-large-language models. In *ICML 2024 Workshop on Trustworthy Multi-modal Foundation Models and AI Agents (TiFA)*.

NVIDIA. 2025. PhysicalAI-Autonomous-Vehicles dataset. https://huggingface.co/datasets/nvidia/PhysicalAI-Autonomous-Vehicles. Hugging Face dataset release.

Chenbin Pan, Burhaneddin Yaman, Tommaso Nesti, Abhirup Mallik, Alessandro G. Allievi, Senem Velipasalar, and Liu Ren. 2024. VLP: Vision language planning for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Faezeh Pasandideh and Achim Rettberg. 2026. LLM-generated fault scenarios for evaluating perception-driven lane following in autonomous edge systems. *arXiv preprint arXiv:2604.07362*.

Qihang Peng, Xuesong Chen, Chenye Yang, Shaoshuai Shi, and Hongsheng Li. 2025. ColaVLA: Leveraging cognitive latent reasoning for hierarchical parallel trajectory planning in autonomous driving. *Preprint*, arXiv:2512.22939.

Kangan Qian, Ziang Luo, Sicong Jiang, Zilin Huang, Jinyu Miao, Zhikun Ma, Tianze Zhu, Jiayin Li, Yangfan He, Zheng Fu, Yining Shi, Boyue Wang, Hezhe Lin, Ziyu Chen, Jiangbo Yu, Xinyu Jiao, Mengmeng Yang, Kun Jiang, and Diange Yang. 2025. FASIONAD++: Integrating high-level instruction and information bottleneck in FAt-Slow fusION systems for enhanced safety in autonomous driving with adaptive feedback. *Preprint*, arXiv:2503.08162.

Kangan Qian, Zhikun Ma, Yangfan He, Ziang Luo, Tianyu Shi, Tianze Zhu, Jiayin Li, Jianhui Wang, Ziyu Chen, Xiao He, Yining Shi, Zheng Fu, Xinyu Jiao, Kun Jiang, Diange Yang, and Takafumi Matsumaru. 2024. FASIONAD: FAst and slow fuSION thinking systems for human-like autonomous driving with adaptive feedback. *Preprint*, arXiv:2411.18013.

Jack Qin, Zhitao Wang, Yinan Zheng, Keyu Chen, Yang Zhou, Yuanxin Zhong, and Siyuan Cheng. 2025. Enhancing end-to-end autonomous driving with risk semantic distillation from VLM. *Preprint*, arXiv:2511.14499.

Ranjan Sapkota, Yang Cao, Konstantinos I. Roumeliotis, and Manoj Karkee. 2025. Vision-language-action (VLA) models: Concepts, progress, applications and challenges. *Preprint*, arXiv:2505.04769.

Tal Schuster, Adam Fisch, Jai Gupta, Mostafa Dehghani, Dara Bahri, Vinh Q. Tran, Yi Tay, and Donald Metzler. 2022. Confident adaptive language modeling. In *Advances in Neural Information Processing Systems*, volume 35.

Lin Sha, Haiyun Guo, Tao Wang, Cong Zhang, Min Huang, Jinqiao Wang, and Qinghai Miao. 2026. ST-Prune: Training-free spatio-temporal token pruning for vision-language models in autonomous driving. *Preprint*, arXiv:2604.19145.

Shuyao Shang, Bing Zhan, Yunfei Yan, Yuqi Wang, Yingyan Li, Yasong An, Xiaoman Wang, Jierui Liu, Lu Hou, Lue Fan, Zhaoxiang Zhang, and Tieniu Tan. 2026. DynVLA: Learning world dynamics for action reasoning in autonomous driving. In *Proceedings of the 43rd International Conference on Machine Learning*.

Hao Shao, Yuxuan Hu, Letian Wang, Guanglu Song, Steven L. Waslander, Yu Liu, and Hongsheng Li. 2024. LMDrive: Closed-loop end-to-end driving with large language models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*.

Ahmed Sharshar, Latif U. Khan, Waseem Ullah, and Mohsen Guizani. 2025. Vision-language models for edge networks: A comprehensive survey. *IEEE Internet of Things Journal*, 12(16):32701–32724.

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

Chonghao Sima, Katrin Renz, Kashyap Chitta, Li Chen, Hanxue Zhang, Chengen Xie, Jens Beißwenger, Ping Luo, Andreas Geiger, and Hongyang Li. 2024. DriveLM: Driving with graph visual question answering. In *Proceedings of the 18th European Conference on Computer Vision*.

Shuhan Tan, Kashyap Chitta, Yuxiao Chen, Ran Tian, Yurong You, Yan Wang, Wenjie Luo, Yulong Cao, Philipp Krähenbühl, Marco Pavone, and Boris Ivanovic. 2026. Latent chain-of-thought world modeling for end-to-end driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Xiaoyu Tian, Junru Gu, Bailin Li, Yicheng Liu, Yang Wang, Zhiyong Zhao, Kun Zhan, Peng Jia, Xianpeng Lang, and Hang Zhao. 2024. DriveVLM: The convergence of autonomous driving and large vision-language models. In *Proceedings of the 8th Conference on Robot Learning (CoRL)*, volume 270 of *Proceedings of Machine Learning Research*, pages 4698–4726.

Fali Wang, Zhiwei Zhang, Xianren Zhang, Zongyu Wu, TzuHao Mo, Qiuhao Lu, Wanjing Wang, Rui Li, Junjie Xu, Xianfeng Tang, Qi He, Yao Ma, Ming Huang, and Suhang Wang. 2024. A comprehensive survey of small language models in the era of large language models: Techniques, enhancements, applications, collaboration with LLMs, and trustworthiness. *Preprint*, arXiv:2411.03350.

Jiancheng Wang, Lidan Liang, Yong Wang, Zengzhen Su, Haifeng Xia, Yuanting Yan, and Wei Wang. 2026a. Multimodal backdoor attack on vlms for autonomous driving via graffiti and cross-lingual triggers. *arXiv preprint arXiv:2604.04630*.

Lu Wang, Tianyuan Zhang, Siyuan Liang, Aishan Liu, Xianglong Liu, Yuwei Chen, and Dacheng Tao. 2025a. Black-box adversarial attack on vision language models for autonomous driving. *arXiv preprint arXiv:2501.13563*.

Xinyang Wang, Qian Liu, Wenjie Ding, Zhao Yang, Wei Li, Chang Liu, Bailin Li, Kun Zhan, Xianpeng Lang, and Wei Chen. 2026b. Unifying language-action understanding and generation for autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yan Wang, Wenjie Luo, Junjie Bai, Yulong Cao, Tong Che, Ke Chen, Yuxiao Chen, Jenna Diamond, Yifan Ding, Wenhao Ding, Liang Feng, Greg Heinrich, Jack Huang, Peter Karkus, Boyi Li, Tsung-Yi Lin, Ming-Yu Liu, Zhijian Liu, Pavlo Molchanov, and 7 others. 2025b. Alpamayo-R1: Bridging reasoning and action prediction for generalizable autonomous driving in the long tail. *Preprint*, arXiv:2511.00088.

Yiru Wang, Zichong Gu, Yu Gao, Anqing Jiang, Zhigang Sun, Shuo Wang, Yuwen Heng, and Hao Sun. 2026c. HiST-VLA: A hierarchical spatio-temporal vision-language-action model for end-to-end autonomous driving. *Preprint*, arXiv:2602.13329.

Yiru Wang, Anqing Jiang, Shuo Wang, Yuwen Heng, Zichong Gu, and Hao Sun. 2026d. ETA-VLA: Efficient token adaptation via temporal fusion and intra-LLM sparsification for vision-language-action models. *Preprint*, arXiv:2603.25766.

Ren Xin, Hongji Liu, Xiaodong Mei, Wenru Liu, Maosheng Ye, Zhili Chen, and Jun Ma. 2025. NetRoller: Interfacing general and specialized models for end-to-end autonomous driving. *Preprint*, arXiv:2506.14589.

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

Minhao Xiong, Zichen Wen, Zhuangcheng Gu, Xuyang Liu, Rui Zhang, Hengrui Kang, Jiabing Yang, Junyuan Zhang, Weijia Li, Conghui He, Yafei Wang, and Linfeng Zhang. 2026. Prune2Drive: A plug-and-play framework for accelerating vision-language models in autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*.

Chejian Xu, Wenhao Ding, Weijie Lyu, Zuxin Liu, Shuai Wang, Yihan He, Hanjiang Hu, Ding Zhao,

and Bo Li. 2022. SafeBench: A benchmarking platform for safety evaluation of autonomous vehicles. In *Advances in Neural Information Processing Systems (NeurIPS), Datasets and Benchmarks Track*.

Haochuan Xu, Yun Sing Koh, Shuhuai Huang, Zirun Zhou, Di Wang, Jun Sakuma, and Jingfeng Zhang. 2025a. Model-agnostic adversarial attack and defense for vision-language-action models. *arXiv preprint arXiv:2510.13237*.

Jiajun Xu, Zhiyuan Li, Wei Chen, Qun Wang, Xin Gao, Qi Cai, and Ziyuan Ling. 2024a. On-device language models: A comprehensive review. *Preprint*, arXiv:2409.00088.

Mingwang Xu, Jiahao Cui, Feipeng Cai, Hanlin Shang, Zhihao Zhu, Shan Luan, Yifang Xu, Neng Zhang, Yaoyi Li, Jia Cai, and Siyu Zhu. 2025b. WAM-Diff: A masked diffusion VLA framework with MoE and online reinforcement learning for autonomous driving. *Preprint*, arXiv:2512.11872.

Yi Xu, Yuxin Hu, Zaiwei Zhang, Gregory P. Meyer, Siva Karthik Mustikovela, Siddhartha Srinivasa, Eric M. Wolff, and Xin Huang. 2024b. VLM-AD: End-to-end autonomous driving through vision-language model supervision. *Preprint*, arXiv:2412.14446.

Yifang Xu, Jiahao Cui, Feipeng Cai, Zhihao Zhu, Hanlin Shang, Shan Luan, Mingwang Xu, Neng Zhang, Yaoyi Li, Jia Cai, and Siyu Zhu. 2025c. WAM-Flow: Parallel coarse-to-fine motion planning via discrete flow matching for autonomous driving. *Preprint*, arXiv:2512.06112.

Zhenhua Xu, Yujia Zhang, Enze Xie, Zhen Zhao, Yong Guo, Kwan-Yee K. Wong, Zhenguo Li, and Hengshuang Zhao. 2024c. DriveGPT4: Interpretable end-to-end autonomous driving via large language model. *IEEE Robotics and Automation Letters*, 9:8186–8193.

Zhenjie Yang, Yilin Chai, Xiaosong Jia, Qifeng Li, Yuqian Shao, Xuekai Zhu, Haisheng Su, and Junchi Yan. 2026. DriveMoE: Mixture-of-experts for vision-language-action model in end-to-end autonomous driving. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10678–10688.

Zhenjie Yang, Xiaosong Jia, Hongyang Li, and Junchi Yan. 2023. LLM4Drive: A survey of large language models for autonomous driving. *Preprint*, arXiv:2311.01043.

Zhaoshu Yu, Bo Wang, Pengpeng Zeng, Haonan Zhang, Ji Zhang, Zheng Wang, Lianli Gao, Jingkuan Song, Nicu Sebe, and Heng Tao Shen. 2025. A survey on efficient vision-language-action models. *Preprint*, arXiv:2510.24795.

Zhenlong Yuan, Chengxuan Qian, Jing Tang, Rui Chen, Zijian Song, Lei Sun, Xiangxiang Chu, Yujun Cai, Dapeng Zhang, and Shuo Li. 2025. AutoDrive-R^2: Incentivizing reasoning and self-reflection capacity for VLA model in autonomous driving. *Preprint*, arXiv:2509.01944.

Dapeng Zhang, Zhenlong Yuan, Zhangquan Chen, Chih-Ting Liao, Yinda Chen, Fei Shen, Qingguo Zhou, and Tat-Seng Chua. 2025. Reasoning-VLA: A fast and general vision-language-action reasoning model for autonomous driving. *Preprint*, arXiv:2511.19912.

Tianyuan Zhang, Lu Wang, Xinwei Zhang, Yitong Zhang, Boyi Jia, Siyuan Liang, Shengshan Hu, Qiang Fu, Aishan Liu, and Xianglong Liu. 2026. Visual adversarial attack on vision-language models for autonomous driving. *Machine Intelligence Research*. ArXiv:2411.18275.

Xingcheng Zhou, Xuyuan Han, Feng Yang, Yunpu Ma, Volker Tresp, and Alois Knoll. 2026. OpenDriveVLA: Towards end-to-end autonomous driving with large vision language action model. In *Proceedings of the AAAI Conference on Artificial Intelligence*.

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

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

Yuxuan Zhu, Shiyi Wang, Wenqing Zhong, Nianchen Shen, Yunqi Li, Siqi Wang, Zhiheng Li, Cathy Wu, Zhengbing He, and Li Li. 2025. A survey on large language model-powered autonomous driving. *Engineering*.