ROADBENCH 2.0: BENCHMARKING MULTIMODAL AGENTS ON END-TO-END URBAN ROAD NETWORK CONSTRUCTION FOR TRAFFIC SIMULATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Large-scale remote-sensing imagery and tool-using agents make automated urban road-network construction increasingly feasible. However, a simulation-ready network must maintain consistency across junction localization, corridor connectivity, road direction, lane structure, and lane-to-lane movements; existing benchmarks typically evaluate these tasks in isolation and do not reveal how errors accumulate through a construction chain. We introduce **RoadBench 2.0**, a dataset and evaluation framework for urban road-network construction from remote-sensing imagery, with aligned OSM references. The dataset contains 166 regional cases covering six US cities, and organizes structural construction and SUMO simulation checks into a connected evaluation protocol, including both stage-wise evaluation with controlled inputs and end-to-end evaluation that passes model predictions downstream. We compare five representative models under a unified protocol. The results show that models can perform well on selected individual stages but remain unreliable across the complete chain, with the highest end-to-end success rate reaching only 0.066. Failure analysis shows that errors in junction localization, corridor connection, and simulation checks accumulate downstream, while ablations reveal stage-dependent effects of OSM. These findings identify cross-stage consistency, error recovery, and simulation feedback as central challenges for simulation-ready road-network construction, and establish **RoadBench 2.0** as a unified basis for studying them.

1 INTRODUCTION

Accurate and up-to-date road networks are fundamental to digital maps, navigation, traffic management, and traffic simulation. Yet constructing a usable road network from large-scale remote-sensing imagery is far more than detecting visible road segments. Junctions, corridor connectivity, travel directions, lane counts, and lane-to-lane movements must form a mutually consistent structure before the resulting network can support routing or simulation (Boeing, 2017; Lopez et al., 2018). In practice, this construction is inherently a multi-stage process: junction predictions determine the endpoints available for corridor construction, road topology and directions constrain reachability, and lane structure further determines feasible movements. Errors introduced upstream can therefore alter subsequent decisions and accumulate throughout the pipeline. The key challenge is not only whether each road-mapping subtask can be solved accurately, but whether these capabilities can be composed over a long horizon into a coherent and executable road network.

Existing research has made substantial progress on individual components of this process, including road extraction and graph construction (Van Etten et al., 2018; Bastani et al., 2018; He et al., 2020a; Hetang et al., 2024), lane-level topology and attributes (Büchner et al., 2023; Wang et al., 2023), and multimodal understanding of urban road scenes (Zhang et al., 2026a). However, these tasks are typically optimized and evaluated separately, often with controlled inputs, leaving error propagation and cross-stage consistency largely unmeasured. In particular, RoadBench evaluates fine-grained road perception and reasoning through tasks such as lane counting, lane designation, network correction, and road-type classification (Zhang et al., 2026a), but does not examine whether such local capabilities can be composed into a complete road-network construction process. Meanwhile, recent multimodal agents increasingly combine visual understanding, geographic information, specialist

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.

Table 1: Feature coverage of representative benchmarks, grouped into road and lane graph construction and urban and geospatial reasoning. Among the benchmarks summarized here, RoadBench 2.0 carries regional road structure through to simulation-ready construction.

		Spatial representation			Interaction and evaluation			
Benchmark	**Input**	**Region graph**	**Lane info.**	**Lane links**	**Tool / action**	**Task decomp.**	**Process eval.**	**Sim. valid.**
SpaceNet (Van Etten et al., 2018)	BEV	✓	—	—	—	—	—	—
UrbanLaneGraph (Büchner et al., 2023)	BEV	✓	✓	✓	—	✓	✓	—
OpenLane-V2 (Wang et al., 2023)	FPV	—	✓	✓	—	✓	—	—
RoadBench (Zhang et al., 2026a)	BEV + FPV	✓	✓	—	—	✓	—	—
CityBench (Feng et al., 2025b)	BEV + FPV	—	—	—	—	✓	—	—
Earth-Bench (Feng et al., 2026)	EO	—	—	—	✓	✓	✓	—
ThinkGeo (Shabbir et al., 2025)	EO (RGB/SAR)	—	—	—	✓	✓	✓	—
RoadBench 2.0	**BEV + OSM**	✓	✓	✓	✓	✓	✓	✓

Region graph includes reconstruction or correction. Lane info.: counts, attributes, or geometry; lane links: explicit lane-to-lane relations. Task decomp.: component-to-composite task organization; process eval.: intermediate artifacts or action traces. Sim. valid.: route-based simulation-readiness assessment rather than graph-only path scoring. BEV: bird's-eye view, including satellite and aerial imagery; FPV: first-person view; EO: Earth observation; OSM: OpenStreetMap.

tools, and multi-step execution (Feng et al., 2026; Xiao et al., 2026; Xia et al., 2026; Wang et al., 2026). These advances make end-to-end automation increasingly feasible, but also expose a harder question for general multimodal agents: *can strong capabilities on individual steps remain reliable when the model must maintain state, propagate its own predictions, and complete a long-horizon structured task?*

To study this problem, we introduce **RoadBench 2.0**, an end-to-end benchmark for evaluating VLM agents on long-horizon construction of simulation-ready urban road networks. We first construct a new manually annotated dataset from the imagery and lane annotations of UrbanLaneGraph (Büchner et al., 2023), adding junction-and-corridor structures, travel directions, lane-to-lane movements, aligned OpenStreetMap references, and simulation-oriented route probes. Based on this dataset, RoadBench 2.0 organizes road-network construction into six dependent stages, from junction localization to SUMO-based simulation readiness. Its evaluation explicitly distinguishes **stage-wise capability** from **end-to-end compositional reliability**: stage-wise evaluation provides reference outputs from preceding stages to isolate individual capabilities, whereas end-to-end evaluation propagates the model's own predictions through the full chain, revealing how upstream errors affect downstream construction. SUMO compilation and required/forbidden route checks further assess whether the resulting network is functionally usable rather than merely structurally plausible. Beyond road-network construction, RoadBench 2.0 provides a structured testbed for measuring and diagnosing the long-horizon reliability of general multimodal agents.

Our contributions are threefold:

- We construct a manually annotated dataset of 166 regional road-network cases, integrating junctions, corridors, travel directions, lane information, lane-to-lane movements, aligned OSM references, and simulation-oriented validation data.
- Based on the dataset, we introduce RoadBench 2.0, an end-to-end benchmark for long-horizon road-network construction with VLM agents, combining controlled stage-wise evaluation, prediction-propagating end-to-end assessment, and SUMO-based functional validation.
- Experiments on representative VLMs reveal a substantial gap between stage-wise capability and end-to-end reliability, with the best success rate reaching only 0.066, and identify distinct failure bottlenecks and error propagation patterns across the construction chain.

2 ROADBENCH 2.0

2.1 BENCHMARK CONSTRUCTION

RoadBench 2.0 starts from the original overhead imagery and lane-level geometry provided by UrbanLaneGraph (Büchner et al., 2023), covering regional scenes across six US cities—Miami, Washington, Austin, Pittsburgh, Palo Alto, and Detroit—at two image resolutions (1600×1600,px and

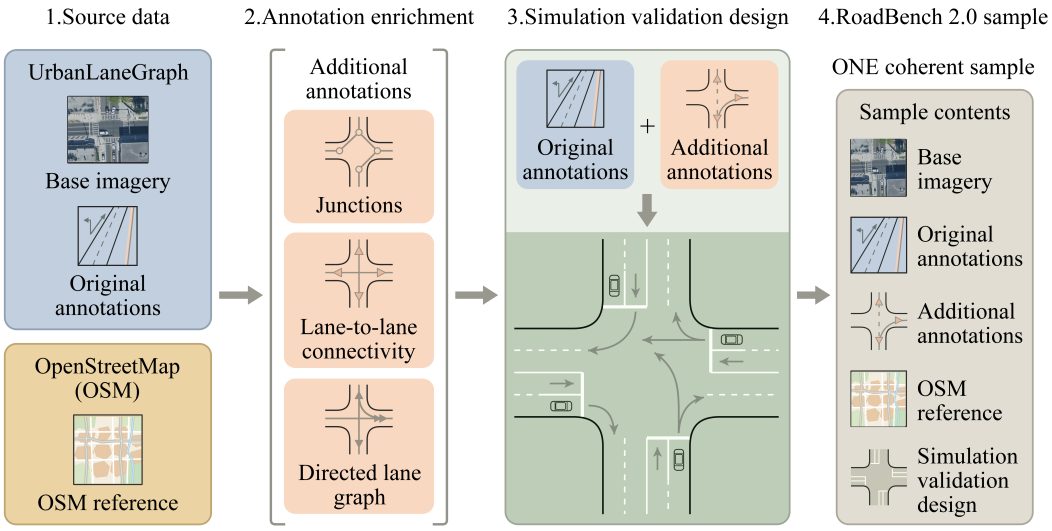

Figure 1: RoadBench 2.0 data construction. The base imagery and lane annotations are retained; road-structure labels are annotated on top; OSM references are aligned to the same frame; and simulation route probes are derived from the annotations.

2500×2500,px), both with a ground sampling distance of 0.15,m/px. What the base lacks is the road-structure layer that a traffic simulator actually consumes—junction topology, corridor connectivity, travel direction, and junction-level lane and movement structure—so we produced that layer by a dedicated annotation campaign rather than by automatic transfer: five annotators worked from a shared written manual that fixes the task region of every case *before* any label is drawn and defines a small closed vocabulary of symbols (junction groups, connection points, boundary ports, visible dead ends, corridors, curve points), so that two annotators looking at the same intersection produce the same kind of object; cases are assigned with overlap across annotators, so a subset of each scene is labeled independently by more than one person and the two versions can be compared, disagreements on topology, geometry, or visibility go to a second reviewer rather than to the original annotator, and every case then passes a nine-point self-check and is accepted as complete, accepted with documented gaps, or excluded with a written reason (details in Appendix C). Figure 1 shows how these pieces are assembled into the released benchmark: the base imagery and lane annotations are kept, the annotated road-structure layer is added on top, the OSM reference is aligned into the same frame, and simulation route probes are derived from the annotations for the functional checks.

2.2 Task Definitions

RoadBench 2.0 organizes network construction into six stages followed by an end-to-end task. The split follows the dependency structure of a routable network: each stage produces a representation that the next stage needs, so a network can only be simulation-ready if every layer is consistent with the layers below it.

T1 localizes junctions, because junctions are the anchors of everything that follows—corridors terminate at them, directions and lane counts attach to their approaches, and movements are defined inside them. **T2** connects those anchors with undirected corridors, producing the connected skeleton: without regional connectivity, local detections do not form a network a vehicle could traverse. **T3** assigns travel direction to each corridor. Direction is what turns an undirected skeleton into a routable graph—without it, reachability is undefined and no route can be planned. **T4** estimates lane counts at each junction approach. Lane counts are the capacity layer that SUMO requires to compile a network, and they also bound the lane indices over which **T5** specifies lane-to-lane movements: which incoming lane may reach which outgoing lane inside a junction. Movements are the finest-grained topology and the last structural layer before function is tested. **T6** then checks that function directly: the predicted network must compile in SUMO and pass required routes while rejecting forbidden ones. This stage tests simulation readiness rather than visual accuracy—a network can match the imagery well and still fail to route.

Finally, **T7** evaluates the full chain end to end. The first six tasks use controlled upstream references so that each transformation can be measured in isolation; T7 instead feeds every stage the model's own predictions, so omissions and errors are free to accumulate. T1–T3 use the added junction, corridor, and direction labels; T4 uses inherited lane labels; and T5 uses movements derived from lanes and junction legs. Figure 2 summarizes the construction stages and evaluation tasks.

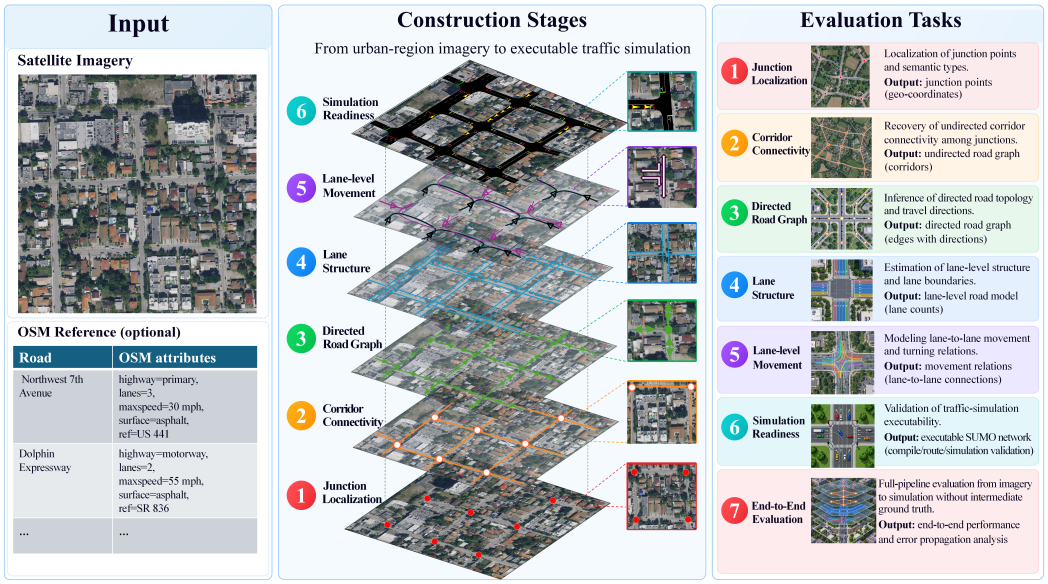

Figure 2: Construction stages and evaluation tasks in RoadBench 2.0. Stages 1–5 produce structural representations, the final stage checks simulation readiness, and the last task evaluates the complete chain.

2.3 Evaluation Framework

RoadBench 2.0 evaluates construction along two complementary views that share the same cases, annotations, and scoring rules, but differ in what each stage is allowed to see (Figure 3). In *controlled-input* evaluation, each stage receives the reference output of every upstream stage together with the aerial image, so a failure at T2 cannot be blamed on a wrong T1; this view answers *can the model perform this transformation when the inputs are correct?*, and T1–T6 are reported under it. In *end-to-end* evaluation the same stages consume the model's own upstream predictions, so errors accumulate from a displaced junction through the corridors it anchors down to the SUMO checks; the aggregate task T7 reports how far a case gets before the chain breaks and whether it ever reaches a usable network, answering *does stage-by-stage capability survive composition?*

The first five stages are scored against the annotations of Section 2.1—junction sets, corridors, travel directions, lane counts, and lane-to-lane movements—using geometric and topological matching rather than identifiers, so a model may use any internal naming or polyline segmentation. The final stage is functional rather than structural: the predicted network must compile in SUMO, accept required routes, and reject forbidden ones. The primary input condition is *Image+OSM*, in which the agent sees the aerial image and an aligned OpenStreetMap extract as a structural prior; a secondary *image-only* condition withholds the prior. In both conditions the agent may consult a library of specialist road- and lane-graph extractors, and an evaluation harness enforces the submission protocol and attempt budgets. Stage metrics and the T7 aggregates are instantiated in Section 3.1.

3 Experiments

3.1 Experimental Design

The benchmark tests one central claim—that simulation-ready road construction is a problem of *composition*, not only of per-stage perception—and the experiments answer three questions: **Q1**,

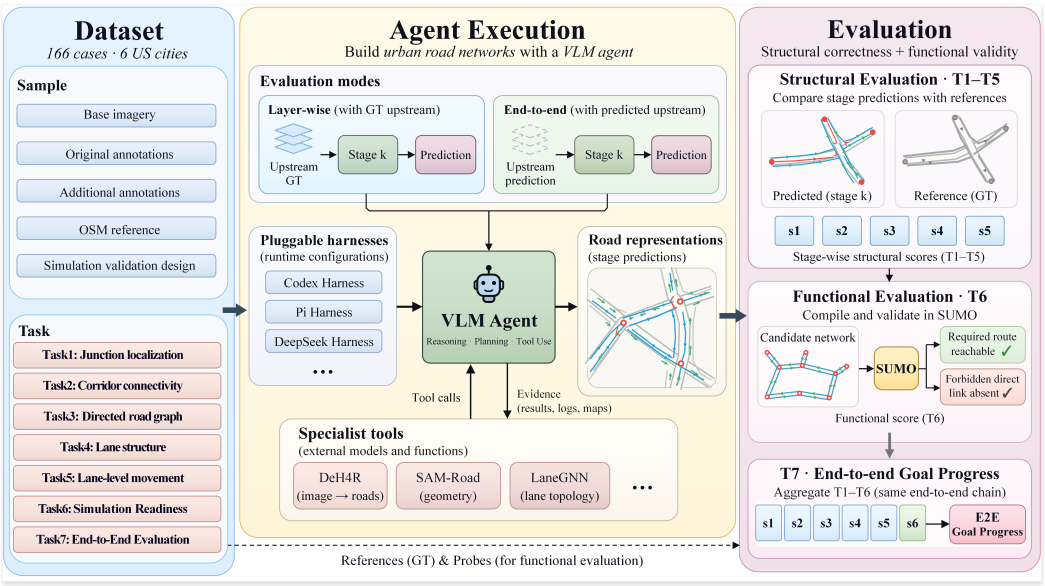

Figure 3: RoadBench 2.0 evaluation framework. Layer-wise evaluation uses upstream references; end-to-end evaluation passes predictions downstream. Structural scores assess T1–T5, SUMO-based route checks assess T6, and T7 aggregates progress.

can models complete each construction stage given correct upstream inputs? **Q2**, does that capability survive when each stage consumes the model's own upstream predictions? and **Q3**, where does the chain break, and what kind of error is responsible? Q1 and Q2 are answered by the two evaluation views of Section 2.3 on the same cases, and Q3 by tracing every chain stage by stage and recording the first failure. We evaluate five representative models—Qwen3.8-27B, DeepSeek-V4.1-Flash, GLM-5.3-Flash, GPT-5.6-Luna, and GPT-6-Astra—under the same harness, specialist-tool library, and input conditions; the first four are agent-oriented foundation models chosen for multi-step planning and tool use, and GPT-6-Astra serves as a strong general-purpose reference. Beyond these questions, a chain ablation on DeepSeek-V4.1-Flash removes the harness, the specialist tools, the OSM prior, or all three while holding the model, cases, and scoring fixed (Section 3.4).

Metrics. Each stage is scored geometrically and topologically against the annotations; identifiers play no role. Table 2 lists the metric set for every task. Notation: G and $\hat{G}$ are the reference and predicted junction sets, and n_G^τ, $n_{\hat{G}}^\tau$ the numbers of reference and predicted junctions whose nearest counterpart is within $\tau = 52.5$ m (350 pixels at 0.15 m/px), computed independently. For T2 and T5, $X \in \{c, a, m\}$ indexes corridor, adjacency, and movement relations, with TP_X, FP_X, FN_X the usual counts. For T3, $\mathcal{Q}$ is the set of reference-reachable ordered junction pairs, $\delta_{uv} = \min(1, |\ell_p(u,v) - \ell_g(u,v)|/\ell_g(u,v))$ with a missing path penalized one, and N_d^+ of N_d matched roads carry the correct direction. For T4, N_l counts reference junction legs with known lane counts, N_l^{exact} exact predictions, and N_l^{pred} legs with a prediction. For T6, N_+ and N_- are positive and negative probes, and N_+^{pass}, N_-^{reject} those that satisfy their requirements; a lower Var. means the two rates are closer and should be read alongside BRA. For T7, each stage score s_k is the mean of its component metrics (T1 NDR/precision, T2 corridor/adjacency F1, T3 APLS/direction, T4 exact, T5 movement F1, T6 BRA), equally weighted, with a wholly missing stage counting zero; the extra metrics (T4 coverage, T5 precision, T6 Var.) are excluded from GP, and SR requires every stage, compilation, and all route probes to pass.

3.2 Main Results

Table 3 reports the seven tasks for all five models, answering Q1 and Q2 side by side: T1–T6 give the controlled-input view of stage capability, and T7 gives the end-to-end view of compositional reliability.

Table 2: Metric set of each task. Structural stages T1–T5 are scored against the annotations by geometric and topological matching; T6 is scored by SUMO compilation and route probes; T7 aggregates stage scores under end-to-end evaluation. Higher is better for all metrics except Var., where lower indicates better balance between the positive-pass and negative-rejection rates.

Task	**Metric set**
T1 junction localization	$\mathrm{NDR}_\tau = n_G^\tau/\lvert G\rvert, \quad \mathrm{Prec}_\tau = n_{\hat{G}}^\tau/\lvert\hat{G}\rvert, \quad \tau = 52.5\,\mathrm{m}$
T2 corridor connectivity	$\mathrm{Prec}_X = \dfrac{TP_X}{TP_X + FP_X}, \quad F_{1,X} = \dfrac{2TP_X}{2TP_X + FP_X + FN_X}, \quad X \in \{c, a\}$
T3 directed road graph	$\mathrm{APLS} = 1 - \dfrac{1}{\lvert\mathcal{Q}\rvert} \sum_{(u,v)\in\mathcal{Q}} \delta_{uv}, \quad \mathrm{DirAcc} = \dfrac{N_d^+}{N_d}$
T4 lane count	$\mathrm{LaneExact} = \dfrac{N_l^{\mathrm{exact}}}{N_l}, \quad \mathrm{Coverage} = \dfrac{N_l^{\mathrm{pred}}}{N_l}$
T5 lane-to-lane movements	Prec_X, $F_{1,X}$ as in T2 with $X = m$
T6 SUMO readiness	$p = \dfrac{N_+^{\mathrm{pass}}}{N_+}, \; n = \dfrac{N_-^{\mathrm{reject}}}{N_-}, \quad \mathrm{BRA}_6 = \dfrac{p+n}{2}, \quad \mathrm{Var}_6 = \dfrac{(p-n)^2}{4}$
T7 end-to-end	$\mathrm{GP} = \dfrac{1}{6}\sum_{k=1}^{6} s_k, \quad \mathrm{SR} = \dfrac{N_{\mathrm{successful}}}{N_{\mathrm{E2E}}}$

Table 3: Main results. T1–T6 use reference upstream outputs; T7 passes model predictions downstream. ↑: higher is better; ↓: lower is better. Best/second-best values are bold/underlined.

Model	**T1** Junction localization		**T2** Corridor connectivity		**T3** Directed road graphs		**T4** Lane-count estimation		**T5** Lane-to-lane movement		**T6** Simulation readiness		**T7** End-to-end construction	
	NDR↑	Prec.↑	Corr.↑	Adj.↑	APLS↑	Dir.↑	Exact↑	Cov.↑	F1↑	Prec.↑	BRA↑	Var.↓	GP↑	SR↑
QWEN-3.8-27B	0.295	<u>0.647</u>	0.330	0.338	0.551	0.770	0.357	0.750	0.364	0.366	0.2214	0.008	0.111	0.006
GLM-5.3-Flash	0.310	0.535	0.412	0.413	0.466	0.778	0.287	0.526	0.494	0.394	0.2967	0.046	0.122	0.006
DeepSeek-V4.1-Flash	<u>0.390</u>	0.582	**0.510**	**0.500**	0.777	**0.817**	<u>0.536</u>	<u>0.964</u>	**0.614**	<u>0.525</u>	**0.3695**	0.045	0.157	0.018
GPT-5.6-Luna	0.311	0.618	0.470	0.459	<u>0.788</u>	0.788	**0.559**	**0.991**	0.295	0.299	0.2550	<u>0.007</u>	<u>0.174</u>	<u>0.048</u>
GPT-6-Astra	**0.392**	**0.696**	<u>0.474</u>	<u>0.462</u>	**0.874**	<u>0.816</u>	0.481	0.951	<u>0.538</u>	**0.534**	<u>0.2987</u>	**0.001**	**0.196**	**0.066**

NDR: neighborhood detection rate. Corr./Adj.: corridor and adjacency F1. Dir.: direction accuracy. Exact/Cov.: lane-count exact accuracy and coverage. BRA is the equally weighted mean of positive-route pass and negative-route rejection rates; Var. is the population variance of these two rates, shown to three decimals (nonzero values below 0.001 are displayed as 0.001), with lower values indicating greater balance. GP/SR: goal progress and end-to-end success rate.

Overall, the five models exhibit distinct capability profiles rather than a single ranking. Astra is the most balanced model: it leads T1, T3 APLS, T5 precision, and T6 route balance, and obtains the highest GP (0.196) and SR (0.066). DeepSeek is strongest on intermediate structural relations, leading T2, T3 direction accuracy, T5 movement F1, and T6 BRA. Luna performs best on lane-count estimation and also ranks second in T3 APLS, GP (0.174), and SR (0.048), while its lower T5 and BRA scores identify later movement and route checking as weaker points. GLM maintains moderate scores across several structural stages and a relatively strong T5 F1, but its lower movement precision and high route variance coincide with a lower GP (0.122) and SR (0.006). Qwen has the second-best T1 precision but a lower NDR, a pattern consistent with conservative predictions that miss many structures; its weaker T2–T6 scores are reflected in the lowest GP (0.111), tied with GLM for the lowest SR (0.006).

Meanwhile, high stage-wise scores do not necessarily translate into high GP or SR. DeepSeek's high T2–T6 scores are obtained under controlled inputs, while its NDR of 0.390 and route variance of 0.045 leave less stable inputs for the chain; Luna's strong T4 result does not compensate for weaker T5 movement quality; GLM's relatively strong T5 F1 is accompanied by lower T5 precision, weaker T4 scores, and the highest route variance among the five models, limiting its downstream reliability; and Qwen's precise but incomplete T1 output limits later construction. SR is conjunctive, so one missing dependency, compilation failure, or route violation makes the complete case unsuccessful. Astra's stronger early localization, competitive downstream structure, and lowest route variance (0.001) jointly support the highest end-to-end scores.

3.3 Failure Mode Analysis

To answer Q3, we trace every chain stage by stage, recording how many cases survive and what kind of error each chain hits first. Figure 4(a) shows Astra carrying nearly the full set through T2 and then losing cases mainly before T3 and again at simulation, whereas DeepSeek drops earlier and more sharply. Figure 4(b) groups these first failures into four types: process/protocol errors, parameter errors, topology/reference errors, and coverage errors. The two models fail in different ways—Astra's failures concentrate in topology and reference mismatches, while DeepSeek's are dominated by parameter errors. Figure 5 shows representative cases for these categories, with ground-truth corridors and junctions against the predicted structure so that the break in the chain is visible on the image.

3.4 Chain Ablation Study

We compare the full chain with four variants that remove the harness, specialist tools, OSM, or all three components on the same cases and protocol, using DeepSeek-V4.1-Flash as the reference model.

The left panel shows a counterintuitive pattern: removing OSM raises GP from 0.157 to 0.224 and SR from 0.018 to 0.030, whereas removing the specialist tools lowers them to 0.131 and 0.012; removing the harness or all three components yields zero for both. Thus, the harness maintains

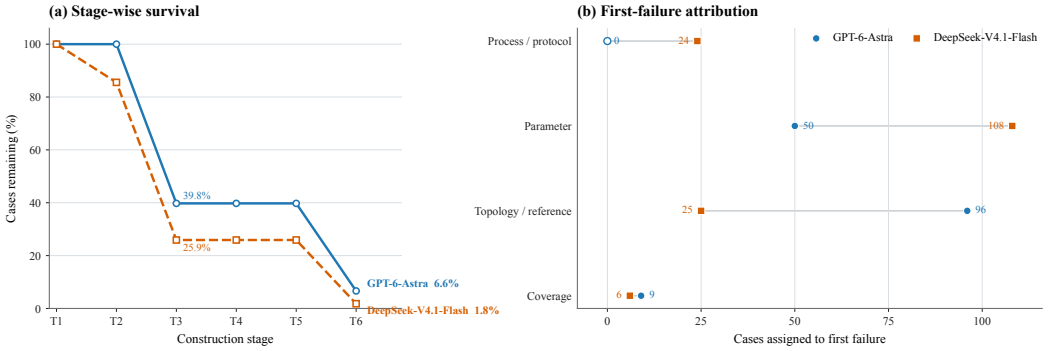

Figure 4: Failure localization for the evaluated chains of GPT-6-Astra and DeepSeek-V4.1-Flash. (a) Percentage of cases remaining after each stage. (b) First-failure counts by error type among failed chains; successful cases are excluded.

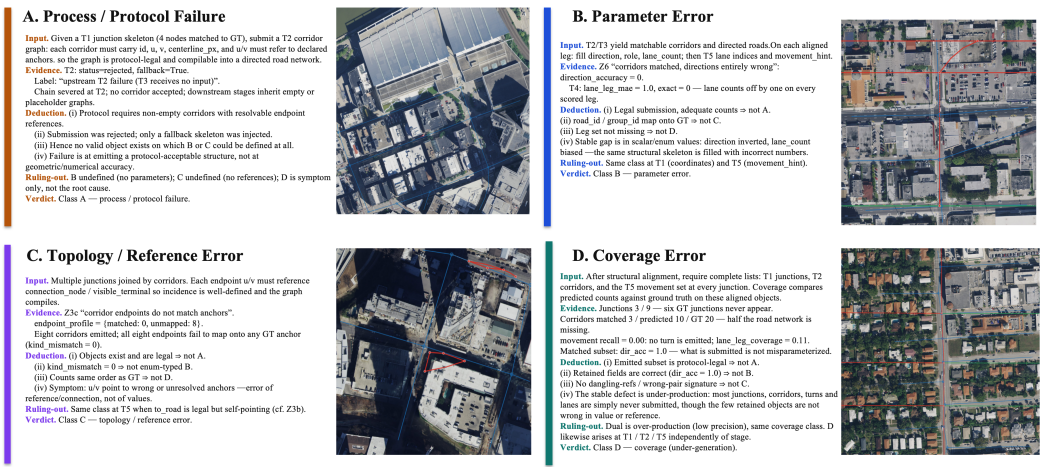

Figure 5: Representative cases for the four first-failure categories: process/protocol, parameter, topology/reference, and coverage errors. Blue denotes ground-truth corridors and junctions, green matched predictions, and red unmatched predictions and predicted junctions.

multi-step state, tools support incremental construction and validation, and OSM can reduce end-to-end reliability under this data distribution.

To localize this effect, the right panel averages the two metrics for T1–T5 and uses BRA for T6, while Var. is read separately. OSM lowers the T1 average from 0.652 to 0.486 but raises T4 and T5 from 0.665 and 0.534 to 0.750 and 0.570; T2–T3 change little, while T6 BRA/Var. shift from 0.384/0.041 to 0.370/0.045. OSM may therefore burden image–map alignment at global junction localization while helping local lane and movement estimation, suggesting stage-selective OSM at T4–T5.

Interestingly, without OSM DeepSeek-V4.1-Flash reaches GP 0.224, 14.3% above GPT-6-Astra's 0.196, but SR 0.030 is only 45.5% of 0.066. GP averages stage progress, whereas SR requires every stage, compilation, and all route probes to pass. The gap shows why end-to-end success requires balanced stage capability and why GP and SR should be read together.

4 Related Work

4.1 Road and Lane Graph Construction

Road reconstruction has progressed from extracting visible roads to predicting connected graph structure. SpaceNet established satellite-image road extraction with path-based evaluation (Van Etten et al., 2018); RoadTracer, VecRoad, and RNGDet grow road graphs iteratively; Sat2Graph predicts a graph-tensor representation; and SAM-Road, SAM-Road++, DeH4R, and RoadGIE extend extraction or graph generation with foundation models and connectivity-aware interaction (Bastani et al., 2018; Tan et al., 2020; Xu et al., 2023; He et al., 2020a; Hetang et al., 2024; Yin et al., 2025; Gong & Ji, 2026; Peng et al., 2026). Other work adds road attributes and lane-level structure: RoadTagger infers lane counts and road types, UrbanLaneGraph and LaneGNN provide overhead imagery aligned with lane graphs, OpenLane-V2 benchmarks lane–lane and lane–traffic-element topology, MapDR associates traffic rules with lane centerlines, and Lanelet2 provides a map representation for geometry, traffic rules, and routing (He et al., 2020b; Büchner et al., 2023; Wang et al., 2023; Chang et al., 2025; Poggenhans et al., 2018).

4.2 Urban Spatial Understanding and Diagnostic Benchmarks

Urban benchmarks evaluate perception and reasoning across modalities and tasks. CityBench combines urban perception with navigation and traffic-control tasks, UrBench covers multi-view localization and scene reasoning, UrbanLLaVA integrates heterogeneous urban modalities, and RoadBench studies lane counting, lane designation, network correction, and road-type classification across bird's-eye, first-person, and cross-view inputs (Feng et al., 2025b; Zhou et al., 2025; Feng et al., 2025a; Zhang et al., 2026a). Diagnostic benchmarks isolate difficulty-scaled spatial reasoning, video-based understanding, and atomic-to-compositional reasoning, while active settings add information gathering and evolving spatial state through urban exploration or cognitive-map evaluation (Wang et al., 2025; Yang et al., 2025; Gong et al., 2026; Zhao et al., 2025; Zhang et al., 2026b).

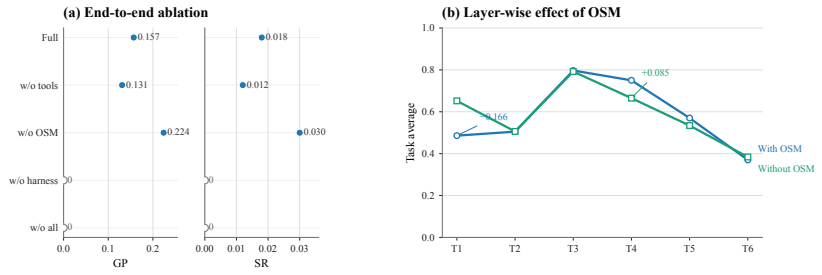

Figure 6: Chain ablation and OSM effects on DeepSeek-V4.1-Flash: (a) GP/SR for full and ablated chains; (b) task averages and BRA, with Var. shown separately.

4.3 Tool-Using Agents and Simulation-Based Evaluation

Tool-using agents combine reasoning with external actions, from the general ReAct formulation to geospatial systems such as Earth-Agent, GeoMMAgent, and MapAgent (Yao et al., 2023; Feng et al., 2026; Xiao et al., 2026; Xia et al., 2026). ThinkGeo, MCP-Bench, ScienceAgentBench, and τ-bench complement this line with evaluations of tool execution, planning, program outcomes, or repeated-trial end states (Shabbir et al., 2025; Wang et al., 2026; Chen et al., 2025; Yao et al., 2025). Simulation provides a further test of downstream usability: nuPlan evaluates planning in closed-loop simulation, SUMO supports network compilation and routing, and ChatSUMO, ChatSUMO-Agent, and TrafficSimAgent automate map, scenario, or simulation workflows (Caesar et al., 2021; Lopez et al., 2018; Li et al., 2024; 2026; Du et al., 2025).

5 Conclusion

RoadBench 2.0 provides a unified evaluation of multimodal agents for simulation-ready urban road construction. The highest end-to-end success rate is 0.066, with failures accumulating from junction localization to simulation and OSM effects varying by stage. These results identify cross-stage consistency and error recovery as key directions for future work.

AI use statement

Generative AI tools assisted with language editing, translation, LaTeX and figure formatting, and manuscript organization. They did not generate or alter data, annotations, results, or citations; the authors verified the experiments and final manuscript and take full responsibility for its content.

References

Favyen Bastani, Songtao He, Sofiane Abbar, Mohammad Alizadeh, Hari Balakrishnan, Sanjay Chawla, Sam Madden, and David DeWitt. Roadtracer: Automatic extraction of road networks from aerial images. In *2018 IEEE/CVF conference on computer vision and pattern recognition*, pp. 4720–4728. IEEE, 2018.

Geoff Boeing. Osmnx: New methods for acquiring, constructing, analyzing, and visualizing complex street networks. *Computers, environment and urban systems*, 65:126–139, 2017.

Martin Büchner, Jannik Zürn, Ion-George Todoran, Abhinav Valada, and Wolfram Burgard. Learning and aggregating lane graphs for urban automated driving. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 13415–13424. IEEE, 2023.

Holger Caesar, Juraj Kabzan, Kok Seang Tan, Whye Kit Fong, Eric Wolff, Alex Lang, Luke Fletcher, Oscar Beijbom, and Sammy Omari. nuplan: A closed-loop ml-based planning benchmark for autonomous vehicles. *arXiv preprint arXiv:2106.11810*, 2021.

Xinyuan Chang, Maixuan Xue, Xinran Liu, Zheng Pan, and Xing Wei. Driving by the rules: A benchmark for integrating traffic sign regulations into vectorized hd map. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6823–6833. IEEE, 2025.

Ziru Chen, Shijie Chen, Yuting Ning, Qianheng Zhang, Boshi Wang, Botao Yu, Yifei Li, Zeyi Liao, Chen Wei, Zitong Lu, et al. Scienceagentbench: Toward rigorous assessment of language agents for data-driven scientific discovery. In *International Conference on Learning Representations*, volume 2025, pp. 96934–96990, 2025.

Yuwei Du, Jun Zhang, Jie Feng, Zhicheng Liu, Jian Yuan, and Yong Li. Trafficsimagent: A hierarchical agent framework for autonomous traffic simulation with mcp control. *arXiv preprint arXiv:2512.20996*, 2025.

Jie Feng, Shengyuan Wang, Tianhui Liu, Yanxin Xi, and Yong Li. Urbanllava: A multi-modal large language model for urban intelligence. In *2025 IEEE/CVF International Conference on Computer Vision (ICCV)*, pp. 6209–6219. IEEE, 2025a.

Jie Feng, Jun Zhang, Tianhui Liu, Xin Zhang, Tianjian Ouyang, Junbo Yan, Yuwei Du, Siqi Guo, and Yong Li. Citybench: Evaluating the capabilities of large language models for urban tasks. In *Proceedings of the 31st ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 5413–5424, 2025b.

Peilin Feng, Zhutao Lv, Junyan Ye, Xiaolei Wang, Xinjie Huo, Jinhua Yu, Wanghan Xu, Wenlong Zhang, Lei Bai, Conghui He, et al. Earth-agent: Unlocking the full landscape of earth observation with agents. In *International Conference on Learning Representations*, volume 2026, pp. 55700–55764, 2026.

Dengxian Gong and Shunping Ji. Deh4r: A decoupled and hybrid method for road network graph extraction. *IEEE Transactions on Geoscience and Remote Sensing*, 64:1–11, 2026.

Ziyang Gong, Wenhao Li, Xianzheng Ma, Songyuan Li, Zhaokai Wang, Songze Li, Jiayi Ji, Xue Yang, Gen Luo, Junchi Yan, and Rongrong Ji. SpaCE-10: A comprehensive benchmark for multimodal large language models in compositional spatial intelligence. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=Df7UjwEgIx.

Songtao He, Favyen Bastani, Satvat Jagwani, Mohammad Alizadeh, Hari Balakrishnan, Sanjay Chawla, Mohamed M Elshrif, Samuel Madden, and Mohammad Amin Sadeghi. Sat2graph: Road graph extraction through graph-tensor encoding. In *European Conference on Computer Vision*, pp. 51–67. Springer, 2020a.

Songtao He, Favyen Bastani, Satvat Jagwani, Edward Park, Sofiane Abbar, Mohammad Alizadeh, Hari Balakrishnan, Sanjay Chawla, Samuel Madden, and Mohammad Amin Sadeghi. Roadtagger: Robust road attribute inference with graph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pp. 10965–10972, 2020b.

Congrui Hetang, Haoru Xue, Cindy Le, Tianwei Yue, Wenping Wang, and Yihui He. Segment anything model for road network graph extraction. In *2024 IEEE/CVF Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, pp. 2556–2566. IEEE, 2024.

Shuyang Li, Talha Azfar, and Ruimin Ke. Chatsumo: Large language model for automating traffic scenario generation in simulation of urban mobility. *IEEE Transactions on Intelligent Vehicles*, 2024.

Shuyang Li, Meng Ma, Talha Azfar, and Ruimin Ke. Chatsumo agent: An llm-based agent for conversational traffic simulation in sumo. *Transportation Research Part C: Emerging Technologies*, 190:105759, 2026.

Pablo Alvarez Lopez, Michael Behrisch, Laura Bieker-Walz, Jakob Erdmann, Yun-Pang Flötteröd, Robert Hilbrich, Leonhard Lücken, Johannes Rummel, Peter Wagner, and Evamarie Wießner. Microscopic traffic simulation using sumo. In *2018 21st international conference on intelligent transportation systems (ITSC)*, pp. 2575–2582. Ieee, 2018.

Chenxu Peng, Chenxu Wang, Yimian Dai, Yongxiang Liu, Ming-Ming Cheng, and Xiang Li. Roadgie: Towards a global-scale aerial benchmark for generalizable interactive road extraction. *arXiv preprint arXiv:2605.26862*, 2026.

Fabian Poggenhans, Jan-Hendrik Pauls, Johannes Janosovits, Stefan Orf, Maximilian Naumann, Florian Kuhnt, and Matthias Mayr. Lanelet2: A high-definition map framework for the future of automated driving. In *2018 21st international conference on intelligent transportation systems (ITSC)*, pp. 1672–1679. IEEE, 2018.

Akashah Shabbir, Muhammad Akhtar Munir, Akshay Dudhane, Muhammad Umer Sheikh, Muhammad Haris Khan, Paolo Fraccaro, Juan Bernabe Moreno, Fahad Shahbaz Khan, and Salman Khan. Thinkgeo: Evaluating tool-augmented agents for remote sensing tasks. *arXiv preprint arXiv:2505.23752*, 2025.

Yong-Qiang Tan, Shang-Hua Gao, Xuan-Yi Li, Ming-Ming Cheng, and Bo Ren. Vecroad: Point-based iterative graph exploration for road graphs extraction. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 8907–8915. IEEE, 2020.

Adam Van Etten, Dave Lindenbaum, and Todd M Bacastow. Spacenet: A remote sensing dataset and challenge series. *arXiv preprint arXiv:1807.01232*, 2018.

Huijie Wang, Tianyu Li, Yang Li, Li Chen, Chonghao Sima, Zhenbo Liu, Bangjun Wang, Peijin Jia, Yuting Wang, Shengyin Jiang, et al. Openlane-v2: A topology reasoning benchmark for unified 3d hd mapping. *Advances in Neural Information Processing Systems*, 36:18873–18884, 2023.

Xingrui Wang, Wufei Ma, Tiezheng Zhang, Celso M de Melo, Jieneng Chen, and Alan Yuille. Spatial457: A diagnostic benchmark for 6d spatial reasoning of large multimodal models. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 24669–24679. IEEE, 2025.

Zhenting Wang, Qi Chang, Hemani Patel, Shashank Biju, Cheng-En Wu, Quan Liu, Aolin Ding, Alireza Rezazadeh, Ankit Parag Shah, Yujia Bao, et al. Mcp-bench: Benchmarking tool-using llm agents with complex real-world tasks via mcp servers. In *International Conference on Learning Representations*, volume 2026, pp. 97377–97425, 2026.

Deguo Xia, Zihan Li, Haochen Zhao, Dong Xie, Yuyao Kong, Xiyan Liu, Jizhou Huang, Mengmeng Yang, and Diange Yang. Mapagent: An industrial-grade agentic framework for city-scale lane-level map generation. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V. 2*, pp. 8289–8300, 2026.

Aoran Xiao, Shihao Cheng, Yonghao Xu, Yexian Ren, Hongruixuan Chen, and Naoto Yokoya. Geommbench and geommagent: Toward expert-level multimodal intelligence in geoscience and remote sensing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2026.

Zhenhua Xu, Yuxuan Liu, Yuxiang Sun, Ming Liu, and Lujia Wang. Rngdet++: Road network graph detection by transformer with instance segmentation and multi-scale features enhancement. *IEEE Robotics and Automation Letters*, 8(5):2991–2998, 2023.

Jihan Yang, Shusheng Yang, Anjali W Gupta, Rilyn Han, Li Fei-Fei, and Saining Xie. Thinking in space: How multimodal large language models see, remember, and recall spaces. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 10632–10643. IEEE, 2025.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. ReAct: Synergizing reasoning and acting in language models. In *International Conference on Learning Representations (ICLR)*, 2023.

Shunyu Yao, Noah Shinn, Pedram Razavi, and Karthik R Narasimhan. τ-bench: A benchmark for tool-agent-user interaction in real-world domains. In *The Thirteenth International Conference on Learning Representations*, 2025.

Pan Yin, Kaiyu Li, Xiangyong Cao, Jing Yao, Lei Liu, Xueru Bai, Feng Zhou, and Deyu Meng. Towards satellite image road graph extraction: A global-scale dataset and a novel method. In *2025 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 1527–1537. IEEE, 2025.

Jun Zhang, Xin Zhang, Jie Feng, Long Chen, Junhui Wang, Zhicheng Liu, Depeng Jin, and Yong Li. Roadbench: Benchmarking mllms on fine-grained spatial understanding and reasoning under urban road scenarios. In *European Conference on Computer Vision*, pp. 177–195. Springer, 2026a.

Pingyue Zhang, Zihan Huang, Yue Wang, Jieyu Zhang, Letian Xue, Zihan Wang, Qineng Wang, Keshigeyan Chandrasegaran, Ruohan Zhang, Yejin Choi, Ranjay Krishna, Jiajun Wu, Li Fei-Fei, and Manling Li. Theory of space: Can foundation models construct spatial beliefs through active exploration? In *International Conference on Learning Representations (ICLR)*, 2026b.

Yong Zhao, Kai Xu, Zhengqiu Zhu, Yue Hu, Zhiheng Zheng, Yingfeng Chen, Yatai Ji, Chen Gao, Yong Li, and Jincai Huang. Cityeqa: A hierarchical llm agent on embodied question answering benchmark in city space. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 12476–12491, 2025.

Baichuan Zhou, Haote Yang, Dairong Chen, Junyan Ye, Tianyi Bai, Jinhua Yu, Songyang Zhang, Dahua Lin, Conghui He, and Weijia Li. Urbench: A comprehensive benchmark for evaluating large multimodal models in multi-view urban scenarios. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 10707–10715, 2025.

A Metric Definitions and Scoring Rules

How matching works. Every structural score is computed by matching predictions to reference outputs geometrically and topologically; identifiers play no role, so a model may use any internal IDs, polyline segmentation, or vertex count. Predictions and references are paired by minimum-cost one-to-one assignment. Scoring runs only inside the public task region, and regions marked unobservable during annotation are excluded from both the numerator and the denominator. Each stage records true/false positives and misses, the valid denominator, and whether the case was scorable, so every number in the paper can be re-derived from the per-case files released with this submission.

T1, junction localization. The main text defines NDR and junction precision at a matching radius of $\tau = 52.5$ m; the two counts are computed independently, each with its own denominator, under one-to-one center matching. Intersections are counted as junction groups, not as internal connection points: a complex intersection with several internal nodes still counts as one junction.

T2, corridor connectivity. Corridor endpoints must first agree with matched anchors (connection points, boundary ports, or visible dead ends); two lines whose endpoints connect differently are not matched on spatial overlap alone. Matched corridors are resampled at 1 m, and overlap is measured with a 3 m buffer: prediction length inside the reference buffer feeds precision, reference length inside the prediction buffer feeds recall. We report length-weighted precision, recall, and F1, together with counts of wrong and missing connections.

T3, directed road graphs. Travel direction must match exactly: a two-way road does not count as one-way, and directions the model cannot support are excluded from the direction score but treated as non-traversable when connectivity is evaluated. Graph paths count as matched when their endpoints correspond and both polylines are covered at least 80% in each direction after resampling. We report node F1, edge/path F1, directed connected-pair accuracy, and a directed APLS-style score computed with one-to-one path matching.

T4, lane counts. Lane counts are scored per travel direction at each junction approach, weighted by matched road length, as exact accuracy, coverage, and mean absolute error. Lanes are indexed from the rightmost lane of the travel direction starting at zero; only physically visible lanes are counted, and "unknown" is recorded explicitly whenever the imagery does not support a number.

T5, lane-to-lane movements. A movement is a connection from one incoming lane to one outgoing lane inside the same junction. Predictions and references are matched after normalizing to these lane pairs. Annotation visibility sets the rules: fully visible approaches form a closed set (an unlisted prediction is an error), partially visible approaches only reward the connections that were confirmed, and unobservable approaches are excluded from scoring. Turn directions (left, straight, right, U-turn) are reported for error analysis and are not part of the F1 key. Legal restrictions and signal policies lie outside the scope of image-based scoring.

T6, simulation readiness. A network passes compilation only if it builds exactly as specified, with no stand-in lanes standing in for unknown ones, and every road has a known lane count. Positive route probes must be routable and arrive; negative probes must be rejected. As stated in the main text, BRA is the equally weighted average of the positive-pass rate and the negative- rejection rate, and Var. is their population variance. Hard failures — vehicles that never arrive, teleports, collisions, wrong-lane insertion — fail the case outright, and compilation success, probe completion, and simulation cleanliness are also reported as separate facts rather than one blended rate.

T7, goal progress and success rate. Each case carries six stage scores from T1 to T6. Goal progress is their average, with a stage that produced no result counting as zero. Success is conjunc-

tive: every stage must pass, the network must compile cleanly, all required probes must behave as expected, the simulation must be clean, and the run must end by completing — not by exhausting its budget, abstaining, or rolling back.

Aggregation. All averages are taken over the same evaluation cases. When a stage produces no result it contributes zero rather than being dropped from the denominator, and placeholder predictions written by the harness after a stage failure are scored low and carry an explicit placeholder flag in the released files. All thresholds were frozen on validation data before any test case was scored.

B How the Agent Is Run: Harness, Prompts, and Tools

B.1 The evaluation harness

Every run is driven by a deterministic harness — a fixed orchestration layer that decides what happens next, while the model only proposes predictions through registered tools. The harness owns attempt budgets, validates every submission against a public schema, accepts or rejects it, records what happened, and stops the run when the budget is gone. It explains public protocol or evidence problems when it rejects a submission, so the model can correct and resubmit, but it never reveals reference answers or scores.

The stages run in order from T1 to T6. In the *controlled-input* evaluation of the main text, the harness inserts the reference output of each finished stage into the prompt of the next one; in the *end-to-end* evaluation it inserts nothing, so every stage lives with whatever the previous stage produced. The model can draw on three kinds of resources: the aerial image (full-image overview and arbitrary crops); the project's specialist tool library — road-graph extractors such as SAM-Road, OpenSerge, RNGDet++, DeH4R and GRASS, the LaneGNN lane-graph extractor, and geometry helpers — which the agent may use at any stage where pixels alone are ambiguous; and, in the assisted condition, the OSM prior block described below. In practice an agent typically consults one or two relevant extractors before submitting a stage, as the stage prompts encourage. Whatever cannot be completed within the attempt budget ends that stage as a rejection or abstention, and every outcome is written to the run log.

B.2 What each ablation removes

The four arms of Section 3.4 map onto simple switches in the released code:

- **No harness:** the agent is replaced by a single direct model call — no tools, no retries, no validation loop.
- **No specialist tools:** only image overview and crop remain; the road and lane extractors are unavailable.
- **No OSM:** the prior block is empty; the image is the only input.
- **None of the three:** all of the above at once.

B.3 Prompt texts (verbatim)

The texts below are reproduced exactly as the model receives them, for reproducibility. Text in brackets is filled in per case (image size, task region, injected upstream outputs, junction-group identifier, movement catalogs). T6 has no free-form prompt — compilation and probing are deterministic — and T7 reuses the same stage prompts in sequence, this time without any injected reference answers.

```
=== System instruction (sent with every stage) ===
You are the RoadAgent image-to-SUMO agent. Use registered dynamic tools.
Codex builtin tools (shell commands, file edits, apply_patch, web access)
are forbidden by the evaluation protocol and count as protocol violations;
never use them, not even for scratch computation.
Professional tools provide fallible evidence; never treat them as truth.
Call professional extractors when the image alone is ambiguous, then
```

```
submit. Do not finish with prose: call the dynamic tool
[submit tool] (canonical name stage.submit_prediction) when ready. Keep
the loop bounded: submit as soon as the structure is consistent; do not
repeat an already completed observation or call tools with no new purpose.

=== Common preamble (appended to every stage) ===
You are evaluating visible physical road structure, not legal traffic
rules. You may call roadagent_image_crop whenever pixels need closer
inspection. Its optional view is raw, direction_detail, or t1_overlay
when available. Use roadagent_image_overview to switch the full image.
Always return to raw when an overlay could conceal lane markings or
traffic evidence. Use source-image pixels with origin at the top-left.
Do not guess occluded facts. The harness may reject roadagent_submit
with public protocol or evidence diagnostics. Correct only those
reported issues and resubmit within the fixed attempt budget. It never
reveals truth or scores. Do not repeat an identical submission. If the
diagnostic names missing evidence, call that missing tool first; once
the required evidence is collected, make one complete submission
immediately.

=== T1: junction localization ===
Task T1 (junction localization only). The only road evidence is the
attached [WxH] aerial image. Locate physical intersections inside this
public task region: [task_region]

Submit a prediction object with exactly these keys:
junction_groups, anchors, corridors, unknown_regions.
- junction_groups: id, center_px, visibility (visible|partial). One
  group per physical intersection.
- anchors: for each junction put one connection_node at the
  intersection center.
- corridors: use [] -- do NOT draw roads in T1.
- unknown_regions: [] unless a material region is truly unknown.

Junction policy (follow strictly):
1. A junction is a physical crossing where >=3 road arms meet, or a
   clear T/Y junction.
2. Do NOT invent a regular grid of points. Equally spaced centers like
   (x,y),(x+d,y),(x+2d,y) are almost always wrong -- discard them.
3. Professional extractors emit many lane-level nodes. Cluster nearby
   nodes into ONE junction candidate; never copy raw tool nodes 1:1.
4. Before submitting, re-check every center on the raw image. Keep only
   pixels that look like a real intersection.
5. Prefer precision over recall when unsure, but you MUST still submit.
   A conservative list of 1-3 clear intersections is valid.

Evidence requirements (hard):
- Call at least one professional road-graph extractor before submit
  (rngdetplusplus preferred; sam_road or openserge also acceptable).
- Also call image.overview at least once (raw view).
- After evidence is available, submit immediately.
- Submit is rejected without professional extract evidence.
- Required shape: junction_groups + anchors + corridors=[] +
  unknown_regions=[].

=== T2: corridor connectivity ===
Task T2 (corridor connectivity only). The attached image remains
available. The following junction_groups and anchors are
evaluator-injected ground truth from T1: [junctions]

Copy rules (violation = ILLEGAL_SUBMISSION):
- Copy every supplied junction and anchor row exactly: same id, same
  center_px, same visibility. Do not add, remove, rename, or move any
  junction or anchor, and reuse the exact ID strings for corridor u/v.
- Do not invent extra terminals unless they are image-boundary
```

```
  endpoints.

Corridor rules:
- corridors: id, u, v, centerline_px, visibility. u/v must be existing
  anchor ids from the copied list; first/last centerline points must
  equal the referenced anchor coordinates exactly.
- Draw roads that visibly connect those fixed junctions; split at
  junction connection nodes. Every junction should have >=2 incident
  corridors when the image shows that connectivity.
- unknown_regions: [] unless a material region is truly unknown.

You may call the SAM-Road or OpenSerge extractors for geometry
evidence, but the junction set is fixed. Hard budget: at most 2
professional extractors and 2 image crops, then submit corridors
immediately.

=== T3: road direction ===
Task T3 [group]. The attached image remains available, and the
following local T1/T2 structure is evaluator-injected ground truth.
Do not add, remove, or redraw T1/T2 elements: [T1/T2]

Submit an object with exactly one key, roads. Create exactly one road
for every supplied T1 corridor. Each road has: id, corridor_id,
direction, critical_anchors_px. direction is u_to_v, v_to_u,
bidirectional, or unknown relative to the corridor's u/v order. Use
bidirectional for ordinary two-way streets even when a median creates
separate carriageways; use unknown only when physical direction cannot
be supported.

Call the LaneGNN lane-graph extractor before submitting this T3 group.
Use a raw image crop when it materially resolves an ambiguous
direction; an explicit abstention is acceptable, but do not skip the
call. If LaneGNN abstains for coverage reasons, use the raw crop and
encode unsupported direction as unknown; do not turn a coverage
limitation into a guessed fact.

=== T4: lane counts (per junction group) ===
Task T4 (lane counts only) for junction group [group]. The attached
image remains available. This local T3 backbone is evaluator-injected
ground truth; it contains no lane-count answers: [T3]
The following junction-leg identities are deterministically implied
for this junction: [legs]

Submit exactly {"junction_legs":[...],"movements":[]}. The harness
owns the protocol version envelope; do not emit a schema field.
movements must be [] -- turns are a separate later task (T5). For
every supplied leg identity, emit exactly one row: copy
junction_group_id, road_id, direction, and role verbatim, then add
lane_count and visibility (visible|partial|unknown). lane_count is the
number of lanes in that travel direction at that junction end; it may
differ at the two ends of one road. Use null only with unknown.

Call the LaneGNN extractor before submitting this T4 group; use a raw
crop when it resolves an ambiguous count. Count only physically
visible lanes; do not invent legal permission, restrictions, or signal
policy.

=== T5: lane-to-lane movements (per junction group) ===
Task T5 (lane-to-lane movements only) for junction group [group]. The
attached image remains available. This local T3 backbone is
evaluator-injected ground truth: [T3]
The junction legs below, including their lane counts, are
evaluator-injected ground truth. Do not re-estimate lane counts; they
bound every lane index you may use: [legs with lane counts]
Movement endpoints use these local catalogs; indices are not lane
```

```
indices: [catalogs]

Submit exactly {"movements":[...]}. Emit neither a schema field nor
junction_legs. Each movement has: id, from_leg_index, from_lane,
to_leg_index, to_lane, movement_hint. Choose indices only from the
given catalogs. Lane indices are zero-based and must stay below the
injected lane_count of the corresponding leg.

Call the LaneGNN extractor before submitting this T5 group. Emit only
physical successors that are visible or strongly supported by road
geometry; prefix every movement id with [group] so merged results stay
unique. Do not invent legal permission, restrictions, access, or
signal policy.
```

B.4 THE OSM PRIOR

In the assisted condition, up to 60 OpenStreetMap highway ways inside the case window are projected into image pixels and appended to the prompt with an explicit disclaimer. The block is introduced as "OSM assistance (public prior for this crop; verify on the image)", and its note reads verbatim:

```
OSM geometry may be stale, incomplete, or shifted versus the aerial
image. Use it as a structural prior only; image pixels decide final
coordinates. Do not copy OSM ids into your submission.
lanes_forward/lanes_backward/turn_lanes/maxspeed/bridge/layer/junction
are OSM tags when present -- verify on the image before relying on
them.
```

The data is labelled "OpenStreetMap highway ways (public prior, not ground truth)". In the image-only condition no block is inserted at all.

C ANNOTATION PROTOCOL AND DATA STATISTICS

All ground truth is drawn by human annotators under a written manual. The task region is fixed during annotation, before any model output for that case is reviewed.

Symbols. Each case uses six symbol types:

Symbol	Meaning
X	junction group — one per physical intersection; this is the unit in which "how many junctions" is counted
N	connection point — where roads actually merge, diverge, or cross; every N belongs to one X
B	boundary port — where a road crosses the task-region boundary
D	visible dead end — where a road genuinely terminates inside the region
C	corridor — one continuous physical carriageway between N/B/D endpoints
P	curve point — geometry only; never a node, endpoint, or junction

Rules annotators follow. A simple cross is one X with one N; divided roads, ramp merges, and roundabouts use one X with several Ns at the real connection points. Roads split at every N they pass through; each physical carriageway gets exactly one C (lane counts are recorded separately, not drawn as parallel lines). Parallel facilities that do not connect — front roads, ramps, overpasses — keep their own C and B. Driving direction may only follow visible evidence such as medians, arrows, or vehicle orientation; when the evidence is missing, the direction is marked uncertain rather than guessed. Lanes are indexed from right to left per travel direction, and every movement records its incoming and outgoing lane pair inside one junction together with a visibility label (fully visible, partly visible, or unobservable). Regions hidden by trees, shadow, cloud, or low resolution are marked unknown with a written reason. Route probes must pass through the target junctions and are used by the evaluator only; they are not part of any model prompt.

Acceptance and quality control. A case is accepted as complete (full stage coverage without critical unknowns), usable with documented gaps, or excluded with a written reason such as an

unresolvable occlusion. Annotators run a nine-point self-check covering the task region, junction counting, endpoint usage, direction and lane evidence, and probe paths; a second reviewer settles any disagreement about topology, geometry, or visibility.

Dataset and OSM reference. The main text reports the benchmark composition (regional cases across six US cities at 1600×1600 and 2500×2500, both at 0.15 m/px). The aligned OSM reference is cut from Geofabrik highway extracts around each case; the assisted condition injects at most 60 in-window ways as described in Appendix B.4. All main results and ablations share one evaluation set — every archived run summary records the same case count. Per-case exclusion lists travel with the supplementary outputs.

D Ablation Results in Full

All five arms of Section 3.4 reuse the same runner, the same protocol, and the same evaluation cases; archived runs confirm that the OSM switch took effect by recording prior coverage of 97.6% with the prior and 0% without it. Table D.1 lists both metrics of every task for all five arms under controlled input. The no-harness and None arms do not execute — without the harness there is no stage loop — so all of their entries are zero, the same zeros plotted in Figure 6(a).

Read together, the table and the discussion above show the load-bearing role of the harness: every other arm still makes measurable progress, while removing the harness empties the scoreboard. Every released case file records, per stage, whether the stage was accepted, incomplete, or rejected, whether a placeholder was written, and the stage metrics of Appendix A; run-level summaries average these files over all cases, which is how the No tools row of Table D.1 was produced. Together with the prompts in Appendix B, the released files are sufficient to re-derive every number in this paper.

	T1		T2		T3		T4		T5		T6	
Arm	NDR$^{\uparrow}$	Prec.$^{\uparrow}$	Corr.$^{\uparrow}$	Adj.$^{\uparrow}$	APLS$^{\uparrow}$	Dir.$^{\uparrow}$	Exact$^{\uparrow}$	Cov.$^{\uparrow}$	F1$^{\uparrow}$	Prec.$^{\uparrow}$	BRA$^{\uparrow}$	Var.$^{\downarrow}$
Full	0.390	0.582	0.510	0.500	0.777	0.817	0.536	0.964	0.614	0.525	0.3695	0.045
w/o tools	0.335	0.524	0.506	0.497	—	—	0.548	0.999	0.606	0.518	0.386	0.038
w/o OSM	0.516	0.787	0.511	0.497	0.773	0.809	0.463	0.869	0.573	0.494	0.384	0.041
w/o harness	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000
None	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000	0.000

Table D.1: Both metrics of every task for all five ablation arms under controlled input. The Full row is the DeepSeek-V4.1-Flash row of Table 3. The No-OSM row is the paired controlled-input comparison on the same cases (one case with incomplete reference annotation is dropped); averaging its two metrics per task reproduces Figure 6(b), and its T6 pair is from Section 3.4. No-tools values are averaged from the released run summaries; em dashes mark cells not reported here. No-harness and None do not execute the pipeline and are zero throughout.