OASIS Solution for AV2 2026 Scenario Mining Challenge

Jeongwoo Park Yuseung Na Seongjae Jeong Minwon Lee Kichun Jo[†]
Hanyang University
{jeongwoopark, yuseungna, sjeong99, minwonlee, kichunjo}@hanyang.ac.kr
[†]Corresponding author

Abstract

This report presents **OASIS**, our solution for the Argoverse 2 (AV2) Scenario Mining (SM) Challenge at the Workshop on Autonomous Driving (WAD), CVPR 2026. The challenge casts scenario mining as natural-language-to-code translation: a large language model (LLM) converts each query into a script of predefined atomic functions that is executed over 3D tracks. We identify four weaknesses of this paradigm: a fixed atomic-function set that covers only 71.7% of test prompts, code-generation hallucination, the inability of code to express visual attributes such as weather or emergency vehicles, and noisy offline-perception trajectories that inflate behavioral false positives. The first three share a single root cause: RefProg relies on one LLM that disregards the prompt's structure and is forced to encode every condition as code. OASIS instead makes the prompt's ontology explicit and routes each sub-condition to the tool best suited to it. It decomposes every query along four axes (Context, Road Actor, Behavior, Relation); three agents progressively narrow the function search space to suppress hallucination; an offline *Augmented aTOMic* (A-TOM) pool raises coverage to ∼96%; a vision-language model (VLM, Qwen3.6 [5]) supplies annotation-absent visual attributes both offline and online; and an Interacting Multiple Model (IMM) smoother [6] refines noisy trajectories. Through these designs, OASIS improves HOTA-Temporal [4] over the RefProg baseline and ranked 1st on the Spatiotemporal Track of the AV2 2026 Scenario Mining Challenge.

1. Introduction

Identifying rare, safety-critical scenarios from massive-scale sensor logs is critical for the development and validation of autonomous-driving systems, yet such scenarios form only a tiny fraction of routine driving data. Traditional scenario mining relies on rigid, hand-authored queries that are hard to scale and cannot interpret nuanced, high-level descriptions of events. The Argoverse 2 Scenario Mining benchmark [2, 1] reframes the task as natural-language-to-code translation: building on 1,000 driving logs with a synchronized HD map, 360° cameras, and LiDAR, it provides 10,000 planning-centric natural-language prompts, and the baseline program-synthesis pipeline, RefProg [1], has an LLM translate each prompt into Python code composed of 28 predefined atomic functions (the logical combinators `scenario_and/or/not` and `output_scenario` are counted separately) that is executed deterministically over the predicted tracks to filter the referred objects.

Although effective, this paradigm has clear weaknesses. (i) The fixed 28-function set covers 96.3% of training prompts but only 71.7% of test prompts, because the test set introduces new ontological content—weather, special infrastructure, emergency vehicles, and ordinal references. (ii) Enumerating the full function list invites hallucination, including non-existent functions/arguments and reversed relational arguments. (iii) Visual attributes such as weather, emergency-vehicle appearance, and pedestrian gesture are absent from the annotations and cannot be expressed by code. (iv) The trajectories produced by offline 3D perception are noisy, so derivative-based behavior recognition produces false positives. We address these with **OASIS**, an ontology-guided agentic scenario-mining system. Our contributions are: a four-axis ontology decomposition with an offline A-TOM function pool that raises coverage to ∼96%; a three-agent retrieval pipeline that narrows the function search space to suppress hallucination; a VLM (Qwen3.6) that supplies annotation-absent visual attributes offline and online; and an IMM smoother that refines noisy trajectories to improve the detection and association accuracy underlying HOTA-Temporal.

2. Method

2.1. Overall Framework

OASIS organizes the whole pipeline around a single four-axis prompt ontology—*Context* (infrastructure and weather), *Road Actor*, *Behavior*, and *Relation*—obtained by decomposing the benchmark prompts. This ontology is the shared backbone of the system: it first guides the offline synthesis of new functions and then guides the online decomposition of each query. Building on it, OASIS keeps the dual-path philosophy of RefProg but reorganizes the system into an *offline* stage that prepares a Scene Database and an augmented

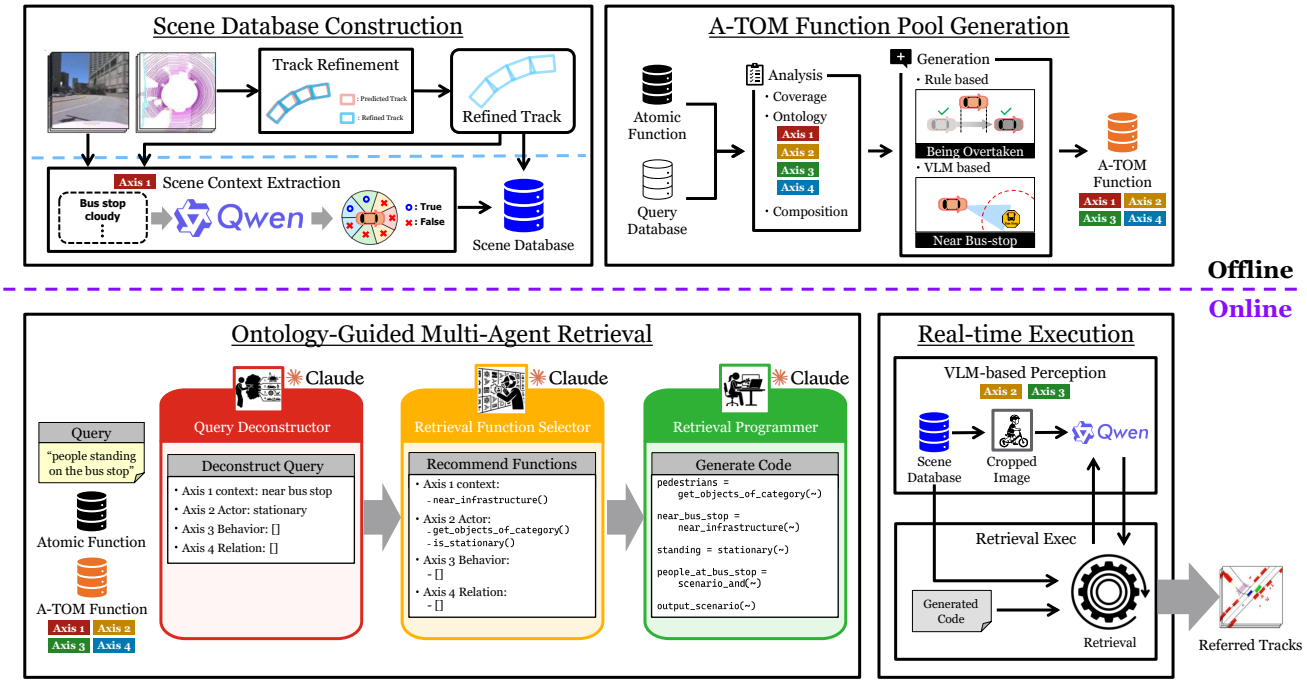

Figure 1. Overall architecture of OASIS. *Offline*: Scene Database construction (Le3DE2E tracks + IMM refinement + VLM-based Scene Context Extraction) and A-TOM function-pool generation. *Online*: ontology-guided multi-agent retrieval (Query Deconstructor → Retrieval Function Selector → Retrieval Programmer) and execution with a Qwen3.6 backend for visual (Actor/Behavior) conditions, producing the referred tracks.

function pool, and an *online* stage that generates and executes retrieval code (Fig. 1). It comprises four components—two offline, two online.

- **Scene Database Construction** (offline, Sec. 2.2). We adopt the official Le3DE2E [3] tracks and refine them with an IMM smoother; in parallel, a VLM (Qwen3.6) performs offline Axis-1 (Context) perception, labeling infrastructure and weather/time-of-day on the multi-view images. The refined tracks and Context labels are stored together in a unified Scene Database.
- **A-TOM Function Pool Generation** (offline, Sec. 2.3). Guided by the four-axis ontology, we analyze the existing atomic functions together with the query database to identify the scenarios the existing functions cannot cover, and synthesize an additional A-TOM function—rule-based or VLM-based—for each gap.
- **Ontology-Guided Multi-Agent Retrieval** (online, Sec. 2.4). Guided by the same ontology and the resulting atomic/A-TOM pool, three agents process the query in sequence—decomposing it along the four axes, selecting the relevant functions, and composing the executable retrieval code.
- **Real-time Execution** (online, Sec. 2.5). The generated code is executed over the Scene Database; Road-Actor/Behavior visual conditions that code cannot resolve are answered online by the VLM on cropped track images, yielding the referred tracks.

2.2. Scene Database Construction

The Scene Database is built from two parallel parts—IMM-based track refinement and VLM-based Context extraction—whose outputs are merged into a single store. The offline 3D perception tracks are noisy at the frame level, and behavior functions that differentiate position, yaw, and velocity amplify this noise into false-positive labels. We therefore refine the official Le3DE2E [3] tracks with an Interacting Multiple Model (IMM) smoother [6] that runs a bank of complementary motion models—constant velocity (CV), constant acceleration (CA), constant turn-rate-velocity (CTRV), and constant turn-rate-acceleration (CTRA)—and derives each track's state by mixing the per-model estimates according to their model probabilities. This adapts to maneuver transitions (cruising, turning, hard braking) while rejecting frame-level noise, which is designed to reduce behavioral false positives and improve the detection accuracy (DetA) and association accuracy (AssA) that compose HOTA-Temporal.

In parallel, a VLM (Qwen3.6) performs offline Axis-1 (Context) perception. For each log it is shown the multi-view camera images and judges, per image, whether each Context attribute is present—infrastructure such as a bus stop, bridge, or parking lot, and weather/time of day such as

Axis	Rule-based A-TOM	VLM-based A-TOM
Context	`within_camera_view`, `in_turn_lane`, `in_parallel_parking`, `near_construction_objects`, `on_median`, `on_road_with_n_lanes`	`is_weather`, `is_time_of_day`, `near_infrastructure`
Road Actor	`active`	`get_visual_actor`
Behavior	`braking`, `braking_hard`, `reversing`, `waiting_to_turn`	`get_visual_behavior`
Relation	`being_overtaken`, `cut_in_front_of`, `between_two_objects`, `group_of`, `nth_object_in_direction`	—
Total	21 new A-TOM functions (16 rule-based, 5 VLM-based)	

Table 1. The 21 A-TOM functions synthesized offline, grouped by ontological axis and by backend. Rule-based functions evaluate deterministic geometric/kinematic conditions; VLM-based functions resolve visually grounded attributes (Context axis served offline from the Scene Database, Road-Actor and Behavior axes served online). Added to the baseline atomic-function set, they raise projected coverage from 71.7% to ~96%.

rain, snow, or night—producing a true/false label. Because these attributes are stable across a log and shared by many prompts, precomputing them amortizes the VLM cost and lets the corresponding Context functions resolve at retrieval time by a cheap lookup. The refined tracks and these Context labels are stored together in a unified Scene Database.

2.3. A-TOM Function Pool Generation

Guided by the four-axis ontology of Sec. 2.1, we analyze the 28 existing atomic functions against the benchmark query distribution. This shows that the test set introduces new Context and Road-Actor content the existing functions cannot express, lowering their coverage to 71.7%. To close this gap, the offline A-TOM (Augmented aTOMic) stage runs a coverage/ontology/composition analysis over the existing functions and the query database and, for each uncovered scenario, synthesizes an additional function along the four axes—either *rule-based* (deterministic conditions such as `being_overtaken()`) or *VLM-based* (visually grounded conditions such as `near_infrastructure()` or `get_visual_actor()`). The 21 resulting A-TOM functions (Table 1) raise the projected coverage to ~96%.

2.4. Ontology-Guided Multi-Agent Retrieval

A single LLM call that sees the entire function library tends to hallucinate. Guided by the four-axis ontology and the atomic/A-TOM pool above, OASIS instead splits retrieval across three specialized agents, all instantiated with Claude. The *Query Deconstructor* maps the prompt onto the same four axes (e.g., "people standing on the bus stop" → Context: *near bus stop*, Actor: *pedestrian, stationary*, Behavior: *none*, Relation: *none*). The *Retrieval Function Selector* then retrieves, per axis, only the few atomic/A-TOM functions the query needs (here `near_infrastructure()` for Context, and `get_objects_of_category()` with `stationary()` for the Road Actor), rather than exposing the full catalog. Finally, the *Retrieval Programmer* composes only those shortlisted functions into an executable script via logical combinators (e.g., `scenario_and`). Restricting each agent's scope—especially the programmer's working set—to the small, axis-aligned shortlist structurally suppresses function/argument hallucination and `track_candidates`/`related_candidates` reversal.

2.5. Real-time Execution

The generated code is executed over the Scene Database. Context-axis conditions resolve by a cheap lookup of the precomputed labels, and precise geometric/kinematic conditions are evaluated deterministically over the refined tracks. Whenever the code references a Road-Actor- or Behavior-axis attribute that deterministic computation cannot resolve—e.g., whether a vehicle is an emergency vehicle (Actor) or whether a pedestrian shows a particular gesture/intent (Behavior)—the executor crops the relevant track region from the camera image and queries the online VLM (Qwen3.6), then merges the answer back into the deterministic candidate filtering. The VLM thus supplies the visual attributes referenced by code while precise geometric conditions remain the responsibility of code, finally yielding the referred tracks.

3. Experiments

3.1. Dataset and Evaluation Metrics

We evaluate on the Argoverse 2 Scenario Mining benchmark [2, 1], which extends the AV2 Sensor dataset (1,000 logs: 700 train / 150 validation / 150 test) with 10,000 planning-centric natural-language queries. The primary metric is **HOTA-Temporal** [4], which jointly assesses detection (DetA) and association (AssA) accuracy for the referred objects over the scenario interval. We also report **HOTA-Track**, **Timestamp Balanced Accuracy** (Timestamp BA), and **Log Balanced Accuracy** (Log BA), which measure classification at the timestamp and full-log (scenario) levels. All agents are accessed through the model API and are driven by Claude Sonnet 4.6 throughout. The VLM

Rank	Team	HOTA-Temporal ↑	HOTA-Track ↑	Timestamp BA ↑	Log BA ↑
Spatiotemporal Track — ranked by HOTA-Temporal					
—	RefProg (baseline) [1]	26.27	36.18	68.07	70.46
1	**HYU_OASIS (ours)**	**38.50**	52.63	74.32	77.12
2	MTL (Argonaut)	37.04	55.11	75.50	80.75
3	MISI (AutoMine)	36.38	49.32	77.21	76.26
Temporal Track — ranked by Timestamp BA					
—	RefProg (baseline) [1]	—	—	68.07	70.46
1	MISI (AutoMine)	—	—	77.21	76.26
2	MTL (Argonaut)	—	—	75.50	80.75
3	**HYU_OASIS (ours)**	—	—	**74.69**	77.84

Table 2. Official CVPR 2026 AV2 Scenario Mining Challenge leaderboard (test split). The challenge has two tracks—the Spatiotemporal Track (ranked by HOTA-Temporal) and the Temporal Track (ranked by Timestamp Balanced Accuracy)—and we list the top three teams on each against the RefProg baseline. OASIS places 1st on the Spatiotemporal Track and 3rd on the Temporal Track. Higher is better for all metrics; "—" marks a metric not ranked on that track. Our two OASIS entries are separate configurations, so their Timestamp BA / Log BA differ between the tracks; on the leaderboard they appear under the team name HYU_OASIS.

is Qwen3.6-35B-A3B [5], served with vLLM [7] on four NVIDIA H100 GPUs.

3.2. Leaderboard Results

The challenge ranks submissions on two tracks: the *Spatiotemporal Track*, ranked by HOTA-Temporal, and the *Temporal Track*, ranked by Timestamp Balanced Accuracy. We submit a configuration tuned for each. OASIS ranks **1st** of all teams on the Spatiotemporal Track (38.50 HOTA-Temporal) and **3rd** on the Temporal Track (74.69 Timestamp BA). Table 2 lists the top three teams on each track against the RefProg baseline.

4. Conclusion

We presented OASIS, our solution for the AV2 2026 Scenario Mining Challenge. Rather than forcing a single LLM to express every condition as code, OASIS makes the prompt ontology explicit and routes each sub-condition to the most suitable tool: a four-axis decomposition with an A-TOM function pool expands coverage to ∼96%; three scoped agents suppress code-generation hallucination; a VLM supplies annotation-absent visual attributes offline and online; and an IMM smoother refines noisy trajectories to improve detection and association accuracy. OASIS improves HOTA-Temporal over the RefProg baseline and ranked 1st on the Spatiotemporal Track of the AV2 2026 Scenario Mining Challenge. Future work includes fine-grained intent/gesture retrieval via trajectory/pose embeddings and a learned fusion of the code and VLM paths.

References

[1] C. Davidson, D. Ramanan, and N. Peri. RefAV: Towards planning-centric scenario mining. *arXiv preprint arXiv:2505.20981*, 2025.

[2] B. Wilson, W. Qi, T. Agarwal, et al. Argoverse 2: Next generation datasets for self-driving perception and forecasting. In *NeurIPS Datasets and Benchmarks*, 2021.

[3] Z. Wang, F. Chen, K. Lertniphonphan, et al. Technical report for Argoverse challenges on unified sensor-based detection, tracking, and forecasting. *arXiv preprint arXiv:2311.15615*, 2023.

[4] J. Luiten et al. HOTA: A higher order metric for evaluating multi-object tracking. *International Journal of Computer Vision*, 2021.

[5] S. Bai, K. Chen, X. Liu, et al. (Qwen Team). Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631*, 2025.

[6] H. A. P. Blom and Y. Bar-Shalom. The interacting multiple model algorithm for systems with Markovian switching coefficients. *IEEE Transactions on Automatic Control*, 1988.

[7] W. Kwon, Z. Li, S. Zhuang, et al. Efficient memory management for large language model serving with PagedAttention. In *ACM SIGOPS 29th Symposium on Operating Systems Principles (SOSP)*, 2023.