VIR-Bench: Evaluating Geospatial and Temporal Understanding of MLLMs via Travel Video Itinerary Reconstruction

Hao Wang[1*], Eiki Murata[2,3*], Lingfang Zhang[1], Ayako Sato[2], So Fukuda[1], Ziqi Yin[1], Wentao Hu[1], Keisuke Nakao[1], Yusuke Nakamura[1], Sebastian Zwirner[1], Yi-Chia Chen[1], Hiroyuki Otomo[2], Hiroki Ouchi[4,2], Daisuke Kawahara[1]

[1]Waseda University
[2]CyberAgent, Inc.
[3]AI Shift, Inc.
[4]Nara Institute of Science and Technology
conan1024hao@akane.waseda.jp, murata_eiki@cyberagent.co.jp, dkw@waseda.jp

Abstract

Recent advances in multimodal large language models (MLLMs) have significantly enhanced video understanding capabilities, opening new possibilities for practical applications. Yet current video benchmarks focus largely on indoor scenes or short-range outdoor activities, leaving the challenges associated with long-distance travel largely unexplored. Mastering extended geospatial-temporal trajectories is critical for next-generation MLLMs, underpinning real-world tasks such as embodied-AI planning and navigation. To bridge this gap, we present **VIR-Bench**, a novel benchmark consisting of 200 travel videos that frames itinerary reconstruction as a challenging task designed to evaluate and push forward MLLMs' geospatial-temporal intelligence. Experimental results reveal that state-of-the-art MLLMs, including proprietary ones, struggle to achieve high scores, underscoring the difficulty of handling videos that span extended spatial and temporal scales. Moreover, we conduct an in-depth case study in which we develop a prototype travel-planning agent that leverages the insights gained from VIR-Bench. The agent's markedly improved itinerary recommendations verify that our evaluation protocol not only benchmarks models effectively but also translates into concrete performance gains in user-facing applications.

Code — https://github.com/nlp-waseda/VIR-Bench
Dataset —
https://soya.infini-cloud.net/share/1302266998c5d047
Extended version — https://arxiv.org/abs/2509.19002

Introduction

Recent advances in multimodal large language models (MLLMs) (Liu et al. 2023; Li et al. 2024b; Lin et al. 2024; OpenAI et al. 2024) have improved remarkable capabilities in video understanding. Lately, research attention has shifted toward evaluating the spatial and temporal reasoning abilities of MLLMs, prompting the proposal of new benchmarks (Grauman et al. 2022; Chandrasegaran et al. 2024; Jia et al. 2024; Yang et al. 2025; Lin et al. 2025). However, existing benchmarks primarily focus on micro-scale scenarios, such as indoor scenes or short-range outdoor activities, leaving macro-scale geospatial scenarios, namely, long-distance travel activities involving multi-day footage across multiple cities, largely unexplored. We argue that long-horizon geospatial-temporal reasoning is essential for next-generation MLLMs, as numerous real-world applications, such as embodied AI planning, navigation, and autonomous driving, heavily rely on these capabilities.

To address this gap, we introduce VIR-Bench, a benchmark to evaluate long-range geospatial-temporal understanding via itinerary reconstruction from travel vlog videos. The core output is a directed visiting order graph (Yamamoto et al. 2025): nodes represent locations at three granularities (prefecture, city, and point of interest (POI)) and edges represent two relations, inclusion for spatial hierarchy and transition for temporal adjacency. By decomposing the task into two sub-tasks: (1) node prediction, identifying all locations visited; and (2) edge prediction, inferring geographic inclusion relations and temporal transition relations among visited locations, VIR-Bench enables separate evaluation of geospatial and temporal intelligence. Because the footage is mostly egocentric or selfie-style, models must construct a holistic understanding from partial views, which further stresses geospatial–temporal reasoning. The dataset comprises 200 travel vlogs filmed across Japan, a major inbound tourism destination, each accompanied by a manually annotated and double-reviewed visiting order graph. We show the overview of VIR-Bench in Figure 1.

Through extensive experiments on state-of-the-art open-weight and proprietary MLLMs, we observe persistent challenge in geospatial and temporal understanding. Particularly, open-weight models suffer from insufficient geographic knowledge and limited capability for long-context reasoning. Although proprietary models achieve better performance, they still struggle on POI node prediction and transition edge prediction, which remain major bottlenecks. Ablations further reveal that more visual context (frames), greater reasoning effort, and access to audio each provide

*These authors contributed equally.

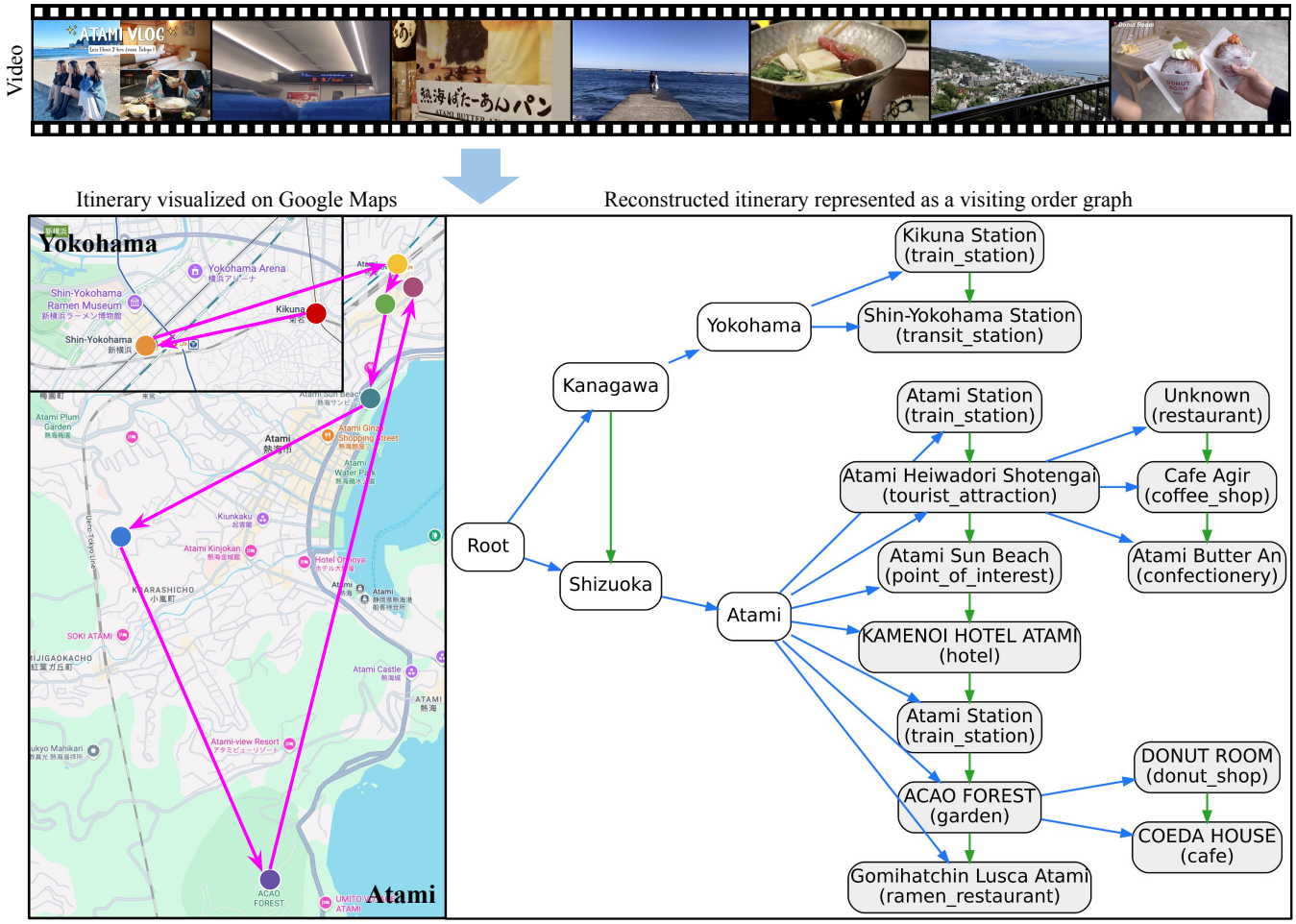


Figure 1: Overview of VIR-Bench. Given an input travel video (**Top**), we reconstruct a visiting order graph (**Right**) whose nodes are visited locations (prefectures, cities, and POIs) and whose edges capture both temporal transitions and geographic containment among the locations. The itinerary visualization (**Left**) omits the second stop at Atami Station for visual clarity. The video frames are adopted from https://www.youtube.com/watch?v=6aJ4CZfn9c8.

consistent gains. Collectively, these findings highlight key obstacles that need to be addressed to advance geospatial-temporal applications in real-world settings.

In addition, we develop a prototype travel-planning agent that generates travel plans directly from videos and their corresponding visiting order graphs. Results from crowdsourcing and automatic evaluations indicate that the itinerary, represented by the POI list, is essential for producing logically sound and feasible plans, underscoring the importance of itinerary reconstruction. Meanwhile, the video provides rich, nuanced context that enhances the attractiveness of a travel plan. A setting that uses both the itinerary and the video leverages these complementary strengths, highlighting a promising approach to generating high-quality travel plans from multimedia sources. These findings validate that our benchmark not only effectively evaluates models but also pushes forward practical user-facing applications.

Related Work

Video Benchmarks

With the rapid advancement of MLLMs, recent video understanding benchmarks have increasingly prioritized evaluating models' spatial and temporal reasoning capabilities. For instance, Ego4D (Grauman et al. 2022) facilitates the evaluation of models' comprehension of past and future events by utilizing curated temporal data, while HourVideo (Chandrasegaran et al. 2024) examines the performance of models in understanding extended-duration video content. VSI-Bench (Yang et al. 2025) assesses a model's ability to infer 3D scene layouts from 2D video inputs, while OST-Bench (Lin et al. 2025) evaluates spatial-temporal understanding by requiring models to explore and interpret information within a 3D space. CityGuessr (Kulkarni, Nayak, and Shah 2024) introduces a video-based benchmark for assessing geo-localization using driving videos, while UrbanVideo-Bench (Zhao et al. 2025) targets the embodied cognitive abilities of MLLMs within urban 3D envi-

ronments using drone-collected footage. Nevertheless, most existing benchmarks primarily feature indoor scenarios or short-distance outdoor movements, lacking extensive long-distance traversal, such as inter-city journeys. Consequently, these benchmarks are insufficient for thoroughly evaluating the geospatial-temporal intelligence of MLLMs. In contrast, VIR-Bench specifically addresses this gap by comprehensively assessing video understanding capabilities across extended spatial (e.g., from Tokyo to Osaka, Hokkaido to Kyoto) and temporal (spanning multiple days) scales.

Itinerary Extraction

Researchers in natural language processing have substantially studied the task of extracting travel trajectories from text (Drymonas and Pfoser 2010; Kaushik et al. 2017; Haris and Gan 2021; Yamamoto et al. 2025). A representative study by Yamamoto et al. (2025) introduces a visiting order graph designed to capture relationships among visited locations and provides a benchmark dataset for training and evaluating itinerary extraction models. In multimodal settings, Pang et al. (2011) proposes a framework aimed at summarizing travelogues by integrating text and images from blogs. More recently, Rosa (2024) leverages MLLMs to perform structured entity extraction from travel videos, while Zhuang et al. (2024) tackles the inverse problem by generating vlogs with diverse travel scenes. This study advances this line of work by providing, to our knowledge, the first systematic investigation into extracting and reconstructing itineraries directly from videos, establishing a new benchmark for video-centric geospatial and temporal understanding.

Itinerary Generation

The complexity of manual trip planning has driven research into automated itinerary generation. Initial approaches were often based on optimization problems like the Tourist Trip Design Problem and classic machine learning (Gavalas et al. 2014; Chen, Ong, and Xie 2016; He, Qi, and Ramamohanarao 2019; Carrillo et al. 2023). More recently, Large Language Models (LLMs) have enabled more sophisticated and flexible frameworks (Chen et al. 2024; Xie et al. 2024). Current research trends include developing novel reasoning paradigms (Gui et al. 2025), creating hybrid systems that combine LLMs with classical planners (de la Rosa et al. 2024), and enhancing personalization through user model integration and interactive feedback (Singh et al. 2024; Chen et al. 2024; Otaki and Baba 2025).

To rigorously evaluate these methods, various benchmarks have been developed. Notable studies include real-world planning (Xie et al. 2024), fine-grained spatio-temporal planning (Chaudhuri et al. 2025), and assessing personalization (Singh et al. 2024).

Existing benchmarks and generation methods primarily use text data such as user preferences and travel logs as input. In contrast, this research uses travel videos as input, aiming to reconstruct the itinerary based on their content.

Dataset Construction

VIR-Bench comprises 200 travel videos filmed across Japan, each paired with a corresponding visiting order graph that captures the itinerary depicted in the video. In this section, we present the construction process of VIR-Bench. We begin by defining the visiting order graph, followed by a detailed description of our data annotation procedure.

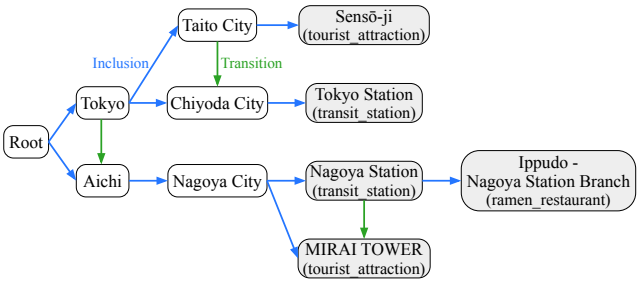


Figure 2: Example of a visiting order graph. **Inclusion edges** represent containment relationships, flowing from a larger geographical area to a smaller one. **Transition edges** indicate chronological movement between distinct locations at the same hierarchical level.

Visiting Order Graph

We adopt and refine the definition of a visiting order graph introduced by Yamamoto et al. (2025). A simplified example is shown in Figure 2.

A visiting order graph is a hierarchical directed graph with four node types:

- **Root node**: the starting node of the graph.
- **Prefecture node**: the highest-level administrative division (e.g., Tokyo, Osaka, Aichi).
- **City node**: a municipality within a prefecture, including Tokyo's special wards, cities, towns, and villages.
- **POI node**: a specific named location (point of interest), such as landmarks, tourist attractions, stations, restaurants, cafes, stores, parks, or museums.

The graph includes two edge types:

- **Inclusion edge**: a directed edge representing containment of one location within another. This edge flows from the larger geographical area to the smaller one.
  - Prefecture → City (e.g., *Aichi* → *Nagoya*).
  - City → POI (e.g., *Nagoya* → *Nagoya Station*).
  - POI → sub-POI (e.g., *Nagoya Station* → *Ippudo - Nagoya Station Branch*).
- **Transition edge**: a directed edge representing movement between two distinct locations at the same hierarchical level; it indicates the chronological flow of travel.
  - Between prefectures (e.g., *Tokyo* → *Aichi*).
  - Between cities within the same prefecture (e.g., *Taito City* → *Chiyoda City*).
  - Between POIs within the same city (e.g., *Nagoya Station* → *MIRAI TOWER*).

To prevent cycles in the graph, we treat multiple visits to the same location as distinct nodes. Following Yamamoto et al. (2025), we also introduce a special "Overlap" edge to

handle POIs that are geographically overlapping but cannot be represented through inclusion edges.

Data Annotation

Identifying locations visited in travel videos is similar to playing GeoGuessr: annotators infer places from visual cues. We recruited 10 Japan-based annotators, each tasked with collecting 20 YouTube travel videos filmed in Japan. The videos could be narrated in English or Japanese. Annotators were asked to identify all visited POIs in each video. We define a "visit" as when the POI appears in the video and it is clear that the videographer visited the facility. For every POI, they recorded the start and end times within the video and provided the corresponding Google Maps URL. When a location could not be uniquely identified (e.g., a cafe shown without its name), they entered the placeholder `UNKNOWN` and recorded the POI category (e.g., `cat_cafe`).

After annotation, we retrieved detailed information of each POI including name, address and categories using Google Places API. Using the timestamped POIs, we then constructed a visiting order graph for each video (e.g., Figure 2). We also conducted a quality check of the generated graphs and corrected errors by rerunning the retrieval step manually when POIs were incorrectly annotated or matched. This pipeline yielded VIR-Bench, a dataset of 200 travel videos (100 in English and 100 in Japanese) paired with their corresponding visiting order graphs. In total, 3,689 POIs were identified across 43 of Japan's 47 prefectures. Detailed annotation guidelines and dataset statistics are provided in the extended version.

Experiments

Task Definition

We aim to generate visiting order graphs directly from videos with MLLMs. However, our preliminary experiments revealed that this end-to-end approach is too difficult for current models. To address this, we decompose the task into two sub-tasks: node prediction and edge prediction. We describe each of these tasks in the following.

Node Prediction This task evaluates models' geospatial understanding, akin to playing "GeoGuessr". Given a video, MLLMs are asked to return all visited locations in three JSON lists (prefectures, cities, and POIs). For each POI, the model must also predict its category.

Edge Prediction Given a video and all visited locations (gold labels, shuffled), MLLMs are asked to predict all inclusion and transition edges that constitute the video's visiting order graph. The output should be a JSON list of tuples formatted as `<source, target, edge_type>`. Inclusion edge prediction evaluates models' geospatial knowledge, while transition edge prediction assesses their temporal understanding. We omit overlap edges in this task due to their low frequency.

Benchmark Models

We evaluate the performance of mainstream MLLMs on VIR-Bench, including both open-weight models (VideoLLaMA3 (Zhang et al. 2025), LLaVA-Video (Zhang et al. 2024)), InternVL3 (Zhu et al. 2025), Qwen2.5-VL (Bai et al. 2025)) and proprietary models (GPT-4.1 (OpenAI 2025a), o4-mini (OpenAI 2025b), Gemini-2.5-Flash and Pro (Comanici et al. 2025)). All models are evaluated in a zero-shot setting. We use as many input frames as permitted by each model's interface or pre-training setup; only the Gemini models accept audio input. Full details appear in the extended version.

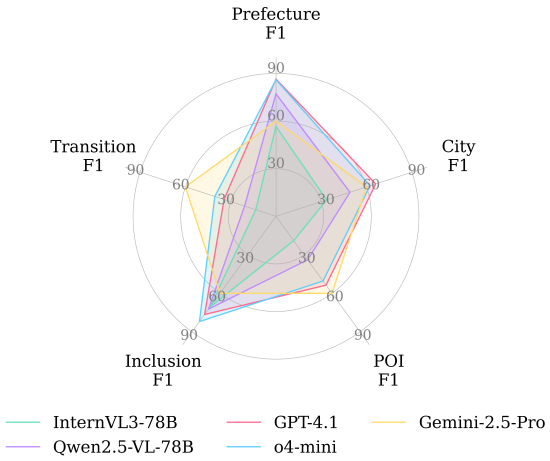


Figure 3: Overall results of top-performing models on VIR-Bench.

Evaluation Metrics

We evaluate models using macro-averaged precision, recall, and F1 across both node and edge prediction. For prefecture and city nodes, a prediction is considered correct only if it exactly matches the gold label's surface name. For POIs, we apply a lightweight sequence-matching algorithm: predictions with a similarity score above 0.7 (high similarity) are treated as correct; predictions with a score above 0.5 (moderate similarity) are also accepted if the predicted POI category matches the gold category; all others are treated incorrect. For inclusion and transition edges, a prediction is counted as correct only when the tuple `<source, target, edge_type>` exactly matches the gold tuple.

Main Results

We present the node-prediction results in Table 1, the edge-prediction results in Table 2, and the overall performance in Figure 3. The overall scores are computed as weighted averages across different node and edge types, with weights proportional to the number of elements in each task category. We also provide a detailed error analysis in the extended version.

Overall Performance Across all five task categories, open-weight models continue to underperform proprietary models. The strongest open model, Qwen2.5-VL-72B, comes close to proprietary performance on the easier categories (prefecture node prediction and inclusion edge pre-

Model	Prefecture			City			POI			OVR
	Precision	Recall	F1	Precision	Recall	F1	Precision	Recall	F1	F1
Open-weight										
VideoLLaMA3-7B	29.8	41.6	31.6	19.1	14.4	14.7	24.0	10.3	13.7	14.6
LLaVA-Video-7B	15.3	13.7	13.2	4.8	6.5	4.9	10.8	4.7	6.0	5.9
LLaVA-Video-72B	17.8	24.7	18.5	10.6	10.4	9.1	10.2	7.9	7.8	8.5
InternVL3-8B	20.1	17.3	17.9	8.8	8.0	7.2	10.4	4.5	6.0	6.8
InternVL3-38B	48.6	46.3	45.7	29.8	19.6	21.8	19.4	11.5	13.6	16.4
InternVL3-78B	58.2	60.5	56.7	41.1	33.7	33.5	30.4	14.8	19.0	22.8
Qwen2.5-VL-7B	46.9	45.1	44.5	30.7	25.3	25.3	27.5	16.8	19.8	21.9
Qwen2.5-VL-32B	74.7	70.6	69.7	53.6	38.1	41.2	37.4	26.1	29.2	33.0
Qwen2.5-VL-72B	**86.2**	**73.6**	**77.2**	**65.4**	**43.8**	**49.0**	**52.3**	**26.6**	**33.9**	38.1
Proprietary										
GPT-4.1	**91.2**	85.9	86.5	**75.9**	62.6	66.0	61.0	51.0	**53.6**	57.0
o4-mini	90.3	85.6	86.1	71.3	59.0	62.3	**63.1**	44.8	50.4	53.9
Gemini-2.5-Flash	88.7	85.5	85.1	74.3	63.7	65.6	57.0	50.4	51.5	55.3
Gemini-2.5-Pro	89.7	**89.0**	**87.7**	73.4	**68.2**	**68.6**	51.8	**58.1**	52.8	57.4

Table 1: Evaluation results on node prediction. "OVR" abbrev for "Overall". The open-weights and proprietary models with the highest and second-highest overall average scores are highlighted with bright green and light green marks.

Model	Inclusion			Transition			OVR
	Precision	Recall	F1	Precision	Recall	F1	F1
Open-weight							
VideoLLaMA3-7B	39.8	31.1	33.4	2.5	1.2	1.4	23.9
LLaVA-Video-7B	22.7	20.7	21.5	1.7	0.9	1.1	15.4
LLaVA-Video-72B	66.3	60.3	62.5	10.3	8.6	8.4	42.4
InternVL3-8B	48.3	45.2	46.0	4.9	2.4	2.5	31.2
InternVL3-38B	64.7	59.7	61.9	15.6	12.3	12.9	41.8
InternVL3-78B	74.4	67.5	70.6	20.7	11.8	13.4	48.9
Qwen2.5-VL-7B	32.0	28.6	29.6	1.5	1.3	1.3	18.0
Qwen2.5-VL-32B	66.6	61.3	63.5	23.5	15.5	16.5	44.6
Qwen2.5-VL-72B	**76.5**	**69.2**	**72.3**	**32.6**	**18.5**	**20.8**	52.4
Proprietary							
GPT-4.1	78.3	75.9	76.5	34.2	36.2	34.4	58.8
o4-mini	86.0	79.0	82.0	40.9	41.0	40.5	64.9
Gemini-2.5-Flash	83.1	74.9	78.5	42.8	42.4	41.7	63.4
Gemini-2.5-Pro	**94.8**	**87.6**	**90.8**	**66.4**	**68.0**	**66.8**	80.7

Table 2: Evaluation results on edge prediction.

diction), but substantial gaps remain on the harder categories (POI node prediction and transition edge prediction). Other open models perform markedly worse: the LLaVA-Video series and InternVL3-8B achieve only single-digit F1 in city and POI node prediction, and five of the nine models also remain in single digits on transition edge prediction. In the proprietary models, Gemini-2.5-Pro is the top performer, especially on edge prediction, yet its F1 scores for city/POI node and transition edge prediction remain around 60. Taken together, these findings indicate that VIR-Bench is highly challenging for current MLLMs and highlight persistent limitations in geospatial and temporal understanding.

Weak Results on Transition Edge Prediction Across the tables, transition edge prediction is the most challenging task. Video-LLaMA3-7B, LLaVA-Video-7B, and Qwen2.5-VL-7B score only around 1, close to random guessing. A plausible factor is the limited number of input frames (e.g., 64 for the LLaVA-Video and InternVL3 series); however, models with larger budgets (180 for Video-LLaMA3 and 256 for Qwen2.5-VL) still struggle, suggesting the issue is not solely input length. Inspecting outputs from lower-performing models reveals two recurrent failure modes: (i) The model sometimes misinterprets the task, including the definitions of inclusion and transition edges; although it produces valid JSON, it yields nonsensical tuples such as `<Tokyo, Shibuya, transition>`. (ii) Transition edges are constrained to connect locations at the same hierarchical level; for POIs, edges are permitted only between POIs within the same city. Models often ignore this constraint and predict cross-city transitions, for example linking a POI in Tokyo to one in Osaka. These factors render the task an even more challenging test of temporal reasoning.

Impact of Model Size When comparing models of different sizes within the same families (LLaVA-Video, In-

Factor	Model	Setting	Node (P/C/POI)	Edge (I/T)
Frames	GPT-4.1	64	85.8 / 62.5 / 39.6	76.6 / 27.6
		128	85.4 / 64.0 / 52.9	**78.8** / 33.5
		256	**86.5 / 66.0 / 53.6**	76.5 / **34.4**
Reasoning	o4-mini	low	**86.8** / 62.0 / 49.1	77.8 / 30.0
		medium	86.1 / 62.3 / 50.4	82.0 / 40.5
		high	86.4 / **63.3 / 51.2**	**83.2 / 43.8**
Audio	Gemini-2.5-Flash	✓	**85.1 / 65.6 / 51.5**	78.5 / **41.7**
		✗	82.5 / 64.0 / 50.5	**82.6** / 22.3

Table 3: Ablation results across frame count, reasoning effort, and audio usage. "Node" = Prefecture / City / POI nodes, and "Edge" = Inclusion / Transition edges. All reported scores are F1 values.

ternVL3, Qwen2.5-VL), we observe steady, scale-driven gains on node prediction, whereas edge prediction shows a sharp jump; for example, transition F1 improves by about $16\times$ from Qwen2.5-VL-7B to Qwen2.5-VL-72B. This pattern reflects the task demands: node prediction is a localized, single-point task that primarily relies on geospatial knowledge encoded in the models, whereas edge prediction requires a holistic view of the itinerary and thus benefits more from larger models with stronger long-context and temporal reasoning. An exception is LLaVA-Video-72B, which shows minimal improvement over LLaVA-Video-7B in POI node prediction. This is likely due to the limited geographic coverage in the LLaVA-OneVision training data (Li et al. 2024a). In contrast, models like Qwen2.5-VL demonstrate strong geo-localization capabilities; in our internal evaluations, Qwen2.5-VL was able to accurately predict POI coordinates, indicating extensive pretraining on geographic data.

Thinking Models' Performance Among the evaluated models, o4-mini and Gemini-2.5-Pro are the only ones that perform explicit "thinking" at inference. Although neither has an available non-thinking counterpart, we use GPT-4.1 and Gemini-2.5-Flash as proximate baselines to gauge what thinking contributes on VIR-Bench. On node prediction, the gains from thinking are limited: for POI nodes, o4-mini achieves higher precision but lower recall, while Gemini-2.5-Pro shows the opposite trend, suggesting different thinking strategies between OpenAI and Google. In contrast, for edge prediction, enabling deliberate thinking yields large gains for both models, especially Gemini-2.5-Pro, indicating that temporal understanding demands more complex reasoning. We presume that Gemini's advantage stems from its use of audio, which supplies continuous, fine-grained temporal cues that sparsely sampled frames cannot provide, highlighting the need for truly multimodal modeling. To further validate the impact of reasoning and audio usage, we conduct additional ablation studies in the following section.

Ablation Study

We conduct additional ablations varying the number of input frames, reasoning effort, and audio usage; results are reported in Table 3. Increasing the number of input frames consistently improves GPT-4.1's overall performance. In particular, the model limited to 64 frames performs poorly on POI node and transition edge prediction, suggesting that for videos in our benchmark, at least 128 frames ($\sim$1 frame every 14s) are a minimum requirement for reliable temporal reasoning. Higher reasoning effort (i.e., longer thinking) leads o4-mini to better performance, especially on transition edge prediction, confirming our earlier observation that the task requires high-level, long-context reasoning. Removing audio from Gemini-2.5-Flash yields worse results across most categories, with nearly a 50% drop on transition edge prediction. This confirms that audio is essential for temporal understanding, as it offers finer and more continuous granularity than the video stream (sampled at 1 fps), likely supporting more consistent long-context reasoning.

Travel-planning Agent

After watching a travel vlog, an animation, or a movie, many fans go on a pilgrimage: visiting the featured locations in the same order as they appear. An automatically generated tranvel plan derived from the video and its visiting order graph would greatly streamline this process.

In this section, we construct MLLM-based agents which aim to provide travel plans given the videos. The purposes of this experiment are (1) to demonstrate the importance of the itinerary reconstruction for this application and (2) to explore the feasibility of generating travel plans from videos, a capability not substantially addressed in prior work.

Task Definition

Input The agent system takes a list of POIs, a video and planning constraints as input. While this list of POIs could be the output of the node prediction task, we use the POI list from the video annotation in this experiment to isolate the evaluation from model performance. The video provides richer information for the planning process. The constraints consist of the number of companions, travel duration and travel budget inspired by previous work (Xie et al. 2024).

Output The output is a travel plan formatted in Markdown. It includes basic information such as the destination, duration, and budget. A core component is a detailed day-by-day itinerary, specifying a schedule of activities, visiting times for each POI, and transportation methods. This itinerary is supplemented with POI details and other rich information extracted from the provided video and/or the search results. Furthermore, the plan provides practical recommendations, including specific restaurants and accommodations with relevant details like price and ratings.

Implementation of the Agent System

We implement the system as a multi-agent framework coordinated by an autonomous orchestrator. This central component is responsible for dynamically determining the execution order of agents, managing the shared state of them.

The framework comprises five specialized agents, each tasked with a specific function: **Plan Agent** constructs the day-by-day schedule, optimizing time allocation based on

the user's budget and constraints. **Google Maps Agent** retrieves POI details. **Route Agent** finds the routes between POIs given the list of POIs. **Accommodation Agent** finds suitable lodging that fits the budget and is optimally located relative to the planned activities. **Summary Agent** integrates the outputs from all other agents to generate a unified final report, including a complete travel plan and a budget breakdown. Each agent can use tools that are appropriate for its purpose, including Google Maps API-based tools and browser-based tools. Further implementation details are shown in the extended version.

Experimental Setup

System Setup To verify the importance of the itinerary reconstruction step, the core task of our benchmark, we prepare 3 input settings: a list of POIs only (**POI**), a video only (**Video**) and both of them (**P+V**). Constraints are always provided as input in all settings.

The backbone models of the orchestrator and all agents are fixed as Gemini-2.5-Pro. For the reproducibility, the temperature is set as zero.

Evaluation Setup We sample 20 pairs of videos and their corresponding annotated graphs as input. Agents under the three settings then generate travel plans for each pair, resulting in 60 plans for evaluation.

We qualitatively evaluate the generated plans with crowdsourcing. Since the videos are filmed across Japan, we hire Japanese-speaking crowdworkers and translate the plans into Japanese. Each plan is evaluated by five workers. The evaluation consists of four tasks: assessing the plan's attractiveness (**Attraction**), verifying the feasibility of the transportation information (**Feasibility**), judging the suitability of the number of POIs (**Density**), and determining the plan's consistency with the video (**Alignment**).

We also evaluate the system's POIs selection. Using GPT-4o (OpenAI 2024), we extract the POIs mentioned in the generated plans and treat them as selected. We then compare selected versus unselected POIs in terms of their on-screen duration in the video and their Google Maps ratings.

Results and Discussion

Crowdsourcing Evaluation The crowdsourcing results are summarized in Figure 4 and Table 4. These results indicate that the P+V setting yielded the most attractive travel plans, achieving the highest average score of 3.73 in the Attraction task (Table 4). This suggests that while a list of POIs provides a solid foundation, the rich information from the video—such as the atmosphere of a place or specific activities shown—is crucial for creating a more appealing plan.

The Alignment task reveals the critical challenge of POI extraction from video (Figure 4d). The video-only setting produced the most polarized results: while it achieved the highest proportion of "mostly aligned" or "aligned" plans (41%), it also generated the largest share of plans deemed "completely unrelated" (31%). This instability suggests that the agent's success is highly dependent on the initial, error-prone step of identifying POIs from raw video. A failure in this stage, as evidenced by the low node prediction F1-scores (Table 1), leads directly to a final plan that is misaligned with the video's content. This underscores that a robust itinerary reconstruction is a foundational prerequisite for generating contextually relevant and reliable travel plans.

Input	Mean Score			Transportation (%)	
	Attract (1-5)	Density (1-5)†	Relevance (1-4)	Has Info	Feasible
POI	3.58	**2.96**	**2.34**	**93.0**	**88.2**
Video	3.46	3.07	2.22	80.0	87.5
P+V	**3.73**	3.13	2.08	89.0	87.6

† For Density: 1=too little, 3=just right, 5=too much.

Table 4: Crowdsourcing results by system configuration.

Input	Duration (seconds)			Google Rating (1-5)		
	Selected	Unselected	Δ	Selected	Unselected	Δ
POI	58.4	36.8	$21.6^{\dagger\dagger\dagger}$	4.29	4.17	$\mathbf{0.12}^{\dagger\dagger\dagger}$
Video	68.6	34.4	$34.2^{\dagger\dagger\dagger}$	4.26	4.19	$0.07^{\dagger}$
P+V	76.3	34.7	$\mathbf{41.7}^{\dagger\dagger\dagger}$	4.25	4.19	$0.06^{\dagger}$
Total	57.2	32.6	$24.6^{\dagger\dagger\dagger}$	4.25	4.17	$0.08^{\dagger\dagger}$

$\dagger : p < 0.05$, $\dagger\dagger : p < 0.01$, $\dagger\dagger\dagger : p < 0.001$

Table 5: Comparison of statistics for selected vs. unselected POIs by system configuration. Δ denotes the difference ("selected" minus "unselected"). In the "Total" row, "selected" refers to POIs chosen in at least one setting, and "unselected" refers to POIs never chosen in any setting.

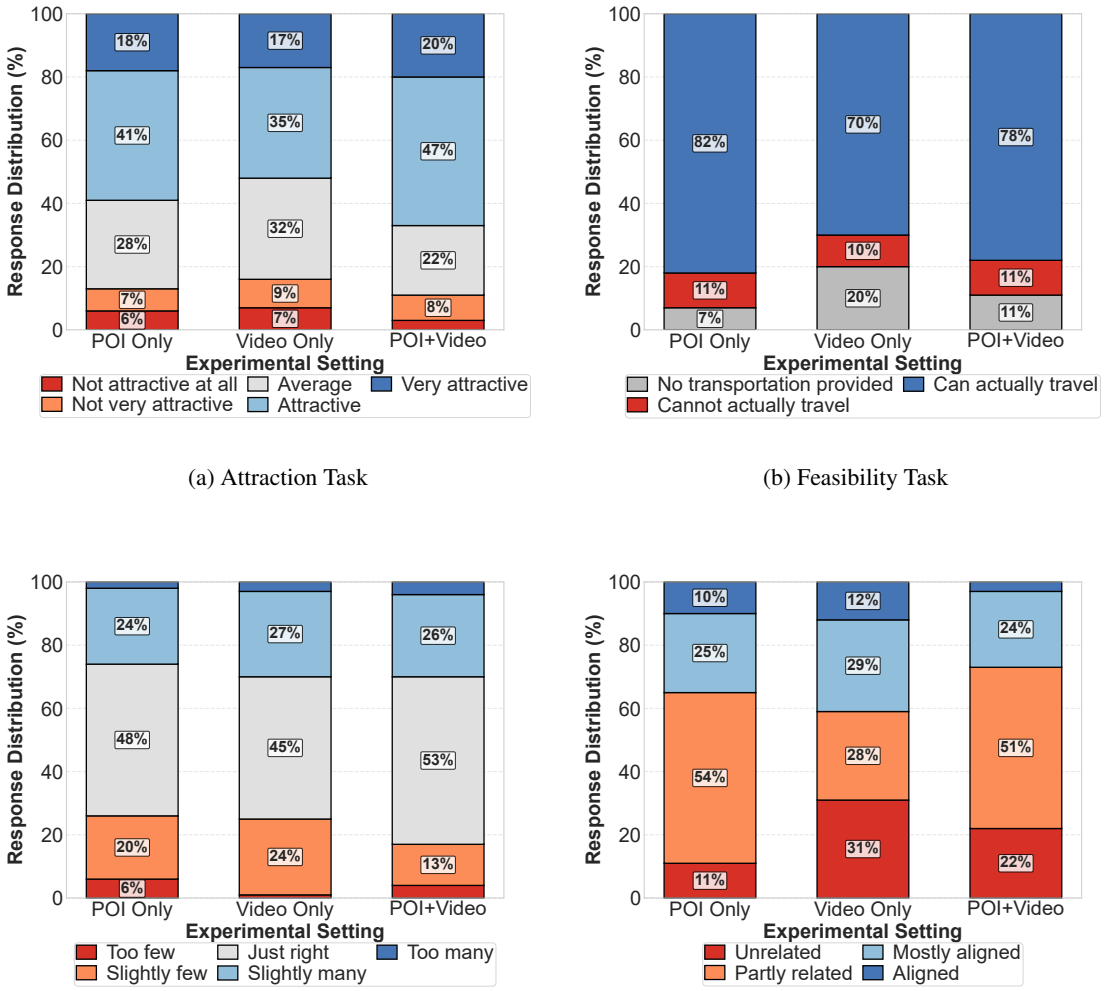
In terms of practicality, all settings generated feasible plans. The POI-only setting was most reliable in providing transportation information, while the video-based settings were slightly better at creating a plan with a "just right" density of activities (Figure 4c). The video-only setting had the highest proportion of plans lacking transportation information (22%, Figure 4b) among the three settings.

Analysis of POI Selection Table 5 provides deeper insights into the agent's underlying POI selection strategy. For all input settings, the agent showed a strong and statistically significant tendency to select POIs that were featured for a longer duration in the video. This effect was most pronounced in the P+V setting ($\Delta = +41.7, p < 0.001$), suggesting that the combination of a POI list and video context enables the agent to most effectively identify and prioritize key locations.

Furthermore, the agent also showed a preference for POIs with higher Google Maps ratings, although this effect was less pronounced than that of video duration. These findings indicate that the agent intelligently synthesizes signals from both the video (visual prominence) and external knowledge sources (user ratings) to make its planning decisions.

Conclusion

We presented VIR-Bench, a video understanding benchmark designed to evaluate long-range geospatial–temporal reasoning through itinerary reconstruction, utilizing visiting or-



(a) Attraction Task

(b) Feasibility Task

(c) Density Task

(d) Alignment Task

Figure 4: Crowdsourcing results of the agent system.

der graphs constructed from 200 travel videos. By decomposing the task into node and edge prediction, we revealed persistent weaknesses in state-of-the-art MLLMs: open-weight models consistently lag behind proprietary ones, and transition edges remain the primary bottleneck. Our prototype travel-planning agent further illustrated the practical value of VIR-Bench, demonstrating that combining POI lists with videos generates the most appealing travel plans, particularly emphasizing visually salient and highly rated POIs. In summary, VIR-Bench provides both a rigorous benchmark and a practical foundation for advancing MLLMs toward video-grounded geospatial–temporal understanding in real-world travel planning scenarios. Looking ahead, we intend to expand our dataset by increasing geographic and linguistic diversity and incorporating a broader range of filming styles. Furthermore, we aim to explore advanced agent systems capable of referencing multiple videos simultaneously, enabling the generation of more engaging travel plans.

License and Access

We release the VIR-Bench dataset strictly for research purposes, in compliance with Article 30-4 (Use for Non-Enjoyment Purposes) and Article 47-5 (Minor Use in Information Analysis Services) of the Japanese Copyright Act. Commercial use of any kind is strictly prohibited. The dataset may not be redistributed on servers outside Japan or under alternative licenses.

References

Bai, S.; Chen, K.; Liu, X.; Wang, J.; Ge, W.; Song, S.; Dang, K.; Wang, P.; Wang, S.; Tang, J.; Zhong, H.; Zhu, Y.; Yang, M.; Li, Z.; Wan, J.; Wang, P.; Ding, W.; Fu, Z.;

Xu, Y.; and 8 others. 2025. Qwen2.5-VL Technical Report. arXiv:2502.13923.

Carrillo, J.; Beltran, V.; Sebastia, L.; and Onaindia, E. 2023. SmartTur+ECO: a conversational recommender system for tourism.

Chandrasegaran, K.; Gupta, A.; Hadzic, L. M.; Kota, T.; He, J.; Eyzaguirre, C.; Durante, Z.; Li, M.; Wu, J.; and Fei-Fei, L. 2024. Hourvideo: 1-hour video-language understanding. *Advances in Neural Information Processing Systems*, 37: 53168–53197.

Chaudhuri, S.; Purkar, P.; Raghav, R.; Mallick, S.; Gupta, M.; Jana, A.; and Ghosh, S. 2025. TripCraft: A Benchmark for Spatio-Temporally Fine Grained Travel Planning. arXiv:2502.20508.

Chen, A.; Ge, X.; Fu, Z.; Xiao, Y.; and Chen, J. 2024. TravelAgent: An AI Assistant for Personalized Travel Planning. arXiv:2409.08069.

Chen, D.; Ong, C.; and Xie, L. 2016. Learning points and routes to recommend trajectories. In *CIKM 2016 - Proceedings of the 2016 ACM Conference on Information and Knowledge Management*, International Conference on Information and Knowledge Management, Proceedings, 2227–2232. United States: Association for Computing Machinery (ACM). Publisher Copyright: © 2016 ACM.; 25th ACM International Conference on Information and Knowledge Management, CIKM 2016 ; Conference date: 24-10-2016 Through 28-10-2016.

Comanici, G.; Bieber, E.; Schaekermann, M.; Pasupat, I.; Sachdeva, N.; Dhillon, I.; Blistein, M.; Ram, O.; Zhang, D.; Rosen, E.; Marris, L.; Petulla, S.; Gaffney, C.; Aharoni, A.; Lintz, N.; Pais, T. C.; Jacobsson, H.; Szpektor, I.; Jiang, N.-J.; and 1 others. 2025. Gemini 2.5: Pushing the Frontier with Advanced Reasoning, Multimodality, Long Context, and Next Generation Agentic Capabilities. arXiv:2507.06261.

de la Rosa, T.; Gopalakrishnan, S.; Pozanco, A.; Zeng, Z.; and Borrajo, D. 2024. TRIP-PAL: Travel Planning with Guarantees by Combining Large Language Models and Automated Planners. arXiv:2406.10196.

Drymonas, E.; and Pfoser, D. 2010. Geospatial route extraction from texts. In *Proceedings of the 1st ACM SIGSPATIAL International Workshop on Data Mining for Geoinformatics*, DMG '10, 29–37. New York, NY, USA: Association for Computing Machinery. ISBN 9781450304306.

Gavalas, D.; Konstantopoulos, C.; Mastakas, K.; and Pantziou, G. 2014. A survey on algorithmic approaches for solving tourist trip design problems. *Journal of Heuristics*, 20(3): 291–328.

Grauman, K.; Westbury, A.; Byrne, E.; Chavis, Z.; Furnari, A.; Girdhar, R.; Hamburger, J.; Jiang, H.; Liu, M.; Liu, X.; Martin, M.; Nagarajan, T.; Radosavovic, I.; Ramakrishnan, S. K.; Ryan, F.; Sharma, J.; Wray, M.; Xu, M.; Xu, E. Z.; and 66 others. 2022. Ego4D: Around the World in 3,000 Hours of Egocentric Video. arXiv:2110.07058.

Gui, R.; Wang, Z.; Wang, J.; Ma, C.; Zhen, H.; Yuan, M.; HAO, J.; Lian, D.; Chen, E.; and Wu, F. 2025. Hyper-Tree Planning: Enhancing LLM Reasoning via Hierarchical Thinking. In *Forty-second International Conference on Machine Learning*.

Haris, E.; and Gan, K. H. 2021. Extraction and Visualization of Tourist Attraction Semantics from Travel Blogs. *ISPRS International Journal of Geo-Information*, 10(10).

He, J.; Qi, J.; and Ramamohanarao, K. 2019. A Joint Context-Aware Embedding for Trip Recommendations. In *2019 IEEE 35th International Conference on Data Engineering (ICDE)*, 292–303.

Jia, B.; Chen, Y.; Yu, H.; Wang, Y.; Niu, X.; Liu, T.; Li, Q.; and Huang, S. 2024. SceneVerse: Scaling 3D Vision-Language Learning for Grounded Scene Understanding. arXiv:2401.09340.

Kaushik, D.; Gupta, S.; Raju, C.; Dias, R.; and Ghosh, S. 2017. Making Travel Smarter: Extracting Travel Information From Email Itineraries Using Named Entity Recognition. In Mitkov, R.; and Angelova, G., eds., *Proceedings of the International Conference Recent Advances in Natural Language Processing, RANLP 2017*, 354–362. Varna, Bulgaria: INCOMA Ltd.

Kulkarni, P. P.; Nayak, G. K.; and Shah, M. 2024. CityGuessr: City-Level Video Geo-Localization on a Global Scale. arXiv:2411.06344.

Li, B.; Zhang, Y.; Guo, D.; Zhang, R.; Li, F.; Zhang, H.; Zhang, K.; Zhang, P.; Li, Y.; Liu, Z.; and Li, C. 2024a. LLaVA-OneVision: Easy Visual Task Transfer. arXiv:2408.03326.

Li, F.; Zhang, R.; Zhang, H.; Zhang, Y.; Li, B.; Li, W.; Ma, Z.; and Li, C. 2024b. LLaVA-NeXT-Interleave: Tackling Multi-image, Video, and 3D in Large Multimodal Models. arXiv:2407.07895.

Lin, J.; Yin, H.; Ping, W.; Molchanov, P.; Shoeybi, M.; and Han, S. 2024. VILA: On Pre-training for Visual Language Models. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 26689–26699.

Lin, J.; Zhu, C.; Xu, R.; Mao, X.; Liu, X.; Wang, T.; and Pang, J. 2025. OST-Bench: Evaluating the Capabilities of MLLMs in Online Spatio-temporal Scene Understanding. *arXiv preprint arXiv:2507.07984*.

Liu, H.; Li, C.; Wu, Q.; and Lee, Y. J. 2023. Visual Instruction Tuning. arXiv:2304.08485.

OpenAI. 2024. GPT-4o System Card. arXiv:2410.21276.

OpenAI. 2025a. Introducing GPT-4.1 in the API. System card / technical report, OpenAI. Release of GPT-4.1 flagship, mini, and nano models with improved instruction-following, long-context capabilities, and lower latency and cost compared to GPT-4o and GPT-4.5.

OpenAI. 2025b. OpenAI o3 and OpenAI o4-mini System Card. System card / technical report, OpenAI. Launch of reasoning models with full tool capabilities including web browsing, Python execution, image and file analysis, image generation, canvas, automations, file search, and memory.

OpenAI; Achiam, J.; Adler, S.; Agarwal, S.; Ahmad, L.; Akkaya, I.; Aleman, F. L.; Almeida, D.; Altenschmidt, J.; Altman, S.; Anadkat, S.; Avila, R.; Babuschkin, I.; Balaji,

S.; Balcom, V.; Baltescu, P.; Bao, H.; Bavarian, M.; Belgum, J.; and 262 others. 2024. GPT-4 Technical Report. arXiv:2303.08774.

Otaki, K.; and Baba, Y. 2025. Travel itinerary recommendation using interaction-based augmented data. *Expert Systems with Applications*, 269: 126294.

Pang, Y.; Hao, Q.; Yuan, Y.; Hu, T.; Cai, R.; and Zhang, L. 2011. Summarizing tourist destinations by mining user-generated travelogues and photos. *Computer Vision and Image Understanding*, 115(3): 352–363. Special issue on Feature-Oriented Image and Video Computing for Extracting Contexts and Semantics.

Rosa, K. D. 2024. Structured Entity Extraction from Travel Videos Using Vision-Language Models. In Neidhardt, J.; Kuflik, T.; Livne, A.; and Zanker, M., eds., *Proceedings of the Workshop on Recommenders in Tourism co-located with the 18th ACM Conference on Recommender Systems (RecSys 2024), Bari, Italy, September 18, 2024*, volume 3886 of *CEUR Workshop Proceedings*, 23–30. CEUR-WS.org.

Singh, H.; Verma, N.; Wang, Y.; Bharadwaj, M.; Fashandi, H.; Ferreira, K.; and Lee, C. 2024. Personal Large Language Model Agents: A Case Study on Tailored Travel Planning. In Dernoncourt, F.; Preoţiuc-Pietro, D.; and Shimorina, A., eds., *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, 486–514. Miami, Florida, US: Association for Computational Linguistics.

Xie, J.; Zhang, K.; Chen, J.; Zhu, T.; Lou, R.; Tian, Y.; Xiao, Y.; and Su, Y. 2024. TravelPlanner: A Benchmark for Real-World Planning with Language Agents. In *Forty-first International Conference on Machine Learning*.

Yamamoto, A.; Otomo, H.; Ouchi, H.; Higashiyama, S.; Teranishi, H.; Shindo, H.; and Watanabe, T. 2025. Graph-Structured Trajectory Extraction from Travelogues. In Che, W.; Nabende, J.; Shutova, E.; and Pilehvar, M. T., eds., *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 14116–14132. Vienna, Austria: Association for Computational Linguistics. ISBN 979-8-89176-251-0.

Yang, J.; Yang, S.; Gupta, A. W.; Han, R.; Fei-Fei, L.; and Xie, S. 2025. Thinking in space: How multimodal large language models see, remember, and recall spaces. In *Proceedings of the Computer Vision and Pattern Recognition Conference*, 10632–10643.

Zhang, B.; Li, K.; Cheng, Z.; Hu, Z.; Yuan, Y.; Chen, G.; Leng, S.; Jiang, Y.; Zhang, H.; Li, X.; Jin, P.; Zhang, W.; Wang, F.; Bing, L.; and Zhao, D. 2025. VideoLLaMA 3: Frontier Multimodal Foundation Models for Image and Video Understanding. arXiv:2501.13106.

Zhang, Y.; Wu, J.; Li, W.; Li, B.; Ma, Z.; Liu, Z.; and Li, C. 2024. Video Instruction Tuning With Synthetic Data. arXiv:2410.02713.

Zhao, B.; Fang, J.; Dai, Z.; Wang, Z.; Zha, J.; Zhang, W.; Gao, C.; Wang, Y.; Cui, J.; Chen, X.; and Li, Y. 2025. UrbanVideo-Bench: Benchmarking Vision-Language Models on Embodied Intelligence with Video Data in Urban Spaces. arXiv:2503.06157.

Zhu, J.; Wang, W.; Chen, Z.; Liu, Z.; Ye, S.; Gu, L.; Tian, H.; Duan, Y.; Su, W.; Shao, J.; Gao, Z.; Cui, E.; Wang, X.; Cao, Y.; Liu, Y.; Wei, X.; Zhang, H.; Wang, H.; Xu, W.; and 32 others. 2025. InternVL3: Exploring Advanced Training and Test-Time Recipes for Open-Source Multimodal Models. arXiv:2504.10479.

Zhuang, S.; Li, K.; Chen, X.; Wang, Y.; Liu, Z.; Qiao, Y.; and Wang, Y. 2024. Vlogger: Make Your Dream A Vlog. arXiv:2401.09414.