MULTI-AGENT GEOSPATIAL COPILOTS FOR REMOTE SENSING WORKFLOWS

Chaehong Lee[†*◇], Varatheepan Paramanayakam[§*], Andreas Karatzas[§], Yanan Jian[†], Michael Fore[†◇], Heming Liao[†], Fuxun Yu[†], Ruopu Li[¶], Iraklis Anagnostopoulos[§], Dimitrios Stamoulis[‡*]
[‡]Department of Electrical and Computer Engineering, The University of Texas at Austin, Austin, TX, USA
[§]School of Electrical, Computer and Biomedical Engineering, Southern Illinois University, Carbondale, IL, USA
[¶]School of Earth Systems and Sustainability, Southern Illinois University, Carbondale, IL, USA
[†]Microsoft Corporation, Redmond, WA, USA – Email: dstamoulis@utexas.edu

***Abstract*—We present GeoLLM-Squad, a geospatial Copilot that introduces the novel *multi-agent* paradigm to remote sensing (RS) workflows. Unlike existing single-agent approaches that rely on monolithic large language models (LLM), GeoLLM-Squad *separates agentic orchestration from geospatial task-solving*, by delegating RS tasks to specialized sub-agents. Built on the open-source AutoGen and GeoLLM-Engine frameworks, our work enables the modular integration of diverse applications, spanning urban monitoring, forestry protection, climate analysis, and agriculture studies. Our results demonstrate that while single-agent systems struggle to scale with increasing RS task complexity, GeoLLM-Squad maintains robust performance, achieving a 17% improvement in agentic correctness over state-of-the-art baselines. Our findings highlight the potential of multi-agent AI in advancing RS workflows.**

***Index Terms*—Geospatial Copilots, Multi-agent systems**

I. Introduction

Remote sensing (RS) workflows are inherently complex, requiring diverse datasets, tools, and analytical methods, all employed with tacit geoscientific expertise [1–5]. Consider a geoscientist: if they use land surface temperature (LST) products or electro-optical (EO) imagery under clear-sky conditions, they might substitute these with ground-station temperature data or synthetic aperture radar (SAR) images when cloud coverage exceeds a threshold. Unfortunately, encoding such nuanced "SAR-over-EO" logic into existing geospatial Copilots requires cumbersome prompting [6]. Monolithic large language models (LLMs) become a bottleneck for achieving meaningful spatiotemporal scale in real-world RS applications due to their finite context windows and token capacities [7]. Multi-agent Copilots have emerged as a novel paradigm, allowing agents to be added to or removed from a team of experts *without* extra prompt tuning, enabling extensibility and simplifying development across numerous tasks [8].

This work aims to harness the untapped potential of multi-agency in RS workflows. We present GeoLLM-Squad, a multi-agent system that *separates agentic orchestration and geospatial task-solving*, capable of scaling to various RS tasks, including urban analysis, forestry, agriculture, and climate studies. Built on the open-source AutoGen [9] and GeoLLM-Engine [5] frameworks, we conduct comprehensive assessment of recent advancements in multi-agent orchestration [8, 10] and workflow-based reasoning [11–14] against human-expert curated workflows. Through this study, we identify key limitations of existing methods in geospatial contexts and propose a hybrid multi-agent approach that outperforms state-of-the-art baselines by up to 17% in agentic correctness.

[*]Equal contribution [◇]Work while at Microsoft; currently new affiliations.

TABLE I
Overview of recent geospatial agent paradigms.

Tool usage	Agency	Application	Methods
Single-turn/tool	Single-agent	Satellite Vision, Urban Map, Agriculture, Forest	[15–21], [22, 23] [24], [25–31]
Multi-turn/tool	Single-agent	Satellite Vision DataOps, Climate	[1–5] [32, 33], [34]
Multi-turn/tool	Multi-agent	Satellite Vision, Map DataOps, Agriculture, Climate, Urban, Forest	**GeoLLM-Squad**

II. Background and Novelty

There is rapid progress in applying geospatial foundation models (GFMs) to RS tasks (Table I). Early approaches follow a single-turn-single-agent paradigm [17, 20], where a fine-tuned LLM performs a narrowly defined task based on predetermined workflows. Recent advancements [5, 33] have introduced multi-turn-single-agent schemes that employ multi-tool APIs to support more complex, long-horizon tasks. However, these systems rely on static workflows (e.g., load-detect-plot "chains" in satellite vision) [5]. More critically, by centralizing all prompting, decision-making, reasoning, and tool execution within a single agent, these methods become bottlenecks when scaling to additional RS tasks [6].

Multi-agent orchestration, such as composition-based [10] and ledger-based scheduling [8], represents a novel agentic paradigm. Through comprehensive experimentation, we identify critical shortcomings in existing approaches when applied to RS workflows: ledger orchestration relies on frequent scheduling GPT calls [7], which becomes prohibitively expensive for RS tasks that involve loading, processing, analyzing, and plotting geospatial data. In contrast, while ledger-free compositional prompting eliminates this overhead, it struggles with error recovery [5]. To address these limitations, GeoLLM-Squad introduces a hybrid approach that combines composition-based reasoning with the iterative ledger-reassessment capabilities, as a "best-of-both-worlds" solution.

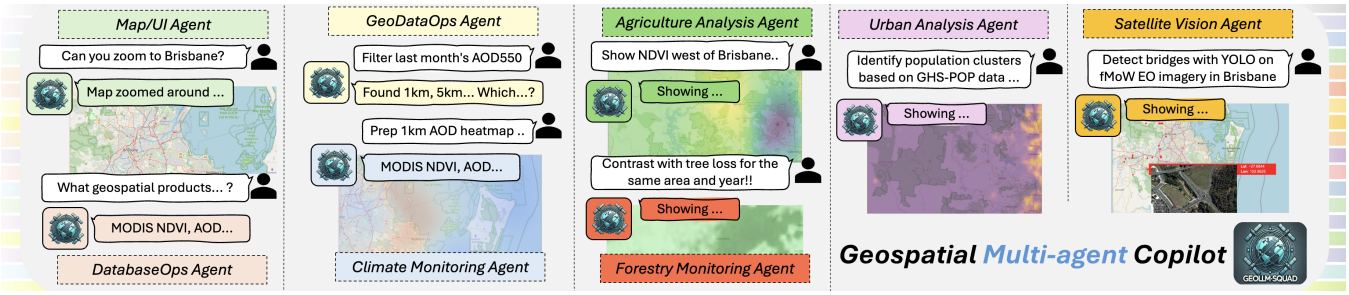

Fig. 1. GeoLLM-Squad is a ***multi-agent*** Copilot for RS workflows that builds on the open-source AutoGen [9] and GeoLLM-Engine [5] frameworks. Our orchestrator supports various geospatial agents and API tools to handle diverse workflows such as climate analysis or urban planning tasks.

TABLE II
OVERVIEW OF GEOSPATIAL PRODUCTS, METRICS, AND DATASETS USED IN GEOLLM-SQUAD WORKFLOWS.

Application	Metric	Dataset/Product	Resolution/Dates	User prompt examples (shortened for illustration)
Agriculture	NDVI, Ref B2	MOD13A3 [35], MOD09GA [36]	1km, 2024	*From NDVI, recommend crop rotation areas in Brisbane*
Climate	LST, AOD550	MYD11A2 [37], MCD19A2 [38]	1km, 2024	*From LST, identify dangerous heatwave regions in Bundaberg*
Urban	Built-S, Popul.	GHS-BUILT-S/POP [39, 40]	3 arcsec, 2020	*From 2020 pop., report overpopulation hotspots for Gympie*
Forest	Canopy, Treeloss	Tree Cover/Loss GFC [41]	1 arcsec, 2020	*From 2020 canopy, recommend reforestation areas in Ipswich*
Vision	Detection, LCC	xView, FAIR, fMoW, ben [42–48]	–	*From xView1, detect and plot aiplanes in Sydney*

III. METHODOLOGY

Agentic Environment. We implement GeoLLM-Squad (Figure 1) on top of two components: GeoLLM-Engine [5] as the agentic frontend and AutoGen as the model serving backend [9]. The frontend integrates interactive map UIs with conversational ChatGPT-like functionality and API tools. The backend enables LLM function-calling for multi-agent communication and orchestration schemes [8, 10].

Geospatial tasks. We consider five RS workflows (Table II) related to urban, agriculture, forest, climate, and satellite-vision studies. For forest data, we utilized tree canopy cover and loss from the Global Forest Change dataset [41]. For urban, we used the (Sentinel-2 and Landsat) GHS-BUILT-S and GHS-POP datasets from the Global Human Settlement layer [39, 40]. Climate workflows employed MODIS Terra products, namely 500m and 1km-resolution Aerosol Optical Depth (AOD 055) and land surface temperature (LST) [38, 37]. Similarly, for agriculture data, we leveraged 1-Month/1km L3 Vegetation Indices and Band 2 surface reflectance [35, 36]. MODIS products were downloaded from NASA Earthdata Search [49] for the entire year 2024. Without loss of generality, our analysis covers the eastern Australian continent. For satellite vision tasks, we used xView1, xView2 (xBD), FAIR1M, xView3, SARFish, fMoW, and BigEarthNet [42–48]. All products (HDF, GeoTIFF) were standardized into GeoPandas formats for frontend-engine integration [5].

Agent tools. We implement tools APIs to emulate functionalities from existing RS single-agents, such as [27, 22, 34], among others. The tools are tested and adapted against in-house workflows, i.e., by injecting status and error messages in their executable code for agentic use. We also repurpose map/UI, database, and data analytics functionalities from [5] as the Map, Database, and DataOps agents, respectively. Overall, GeoLLM-Squad comprise a total of 521 API functions, nearly a 3× increase compared to single-agent baselines [5, 3].

Task generation. Following [12], we provide the latest GPT-4o [51] with user-task "seeds" and generate 56 sample prompts (7 per agent). We then construct human-curated solutions outlining the steps required to complete each sample. Next, we augment an "oracle" GPT-4o with these annotated solutions via RAG [14], prompting it to generate "noisy" ground-truths, which are then inspected by human annotators and the correct solutions are recorded as "*gold*" ground-truths. We note that this process is the only offline step necessitating human annotation. We then use the human-vetted solutions and via few-shot "training" we prompt the oracle GPT-4o to generate 250 *new* tasks (prompts) per agent along with their corresponding solutions, for a total of 2k tasks. This process results in representative "*pseudo*" datasets that reflects realistic workflows as shown in [5, 52, 53].

Orchestrator. Our scheduling follows compositional reasoning [10] to devise program-like schedules in natural language (see below), which specify the sequential set of agents to execute the task and their (sub)prompts. Execution follows the prescribed schedule, and the Orchestrator aggregates return messages to check task completion. If incomplete, the schedule is revised and the loop is repeated; otherwise, the final response and the updated UI/map are returned to the user.

Agents. Each (sub)agent completes its assigned (sub)task using its dedicated toolkit via function calling. We note the advantage of having multiple experts; for example, the agriculture agent's action space includes tools for crop rotation recommendations based on low-NDVI clusters without requiring to explicitly handle NDVI data loading.

TABLE III
AGENTIC PERFORMANCE EVALUATION AND DOWNSTREAM METRICS ACROSS RS TASKS WITH GPT-4O-MINI [50].

Method	Prompting		Performance		Agriculture		Climate		Urban		Forestry		Vision	
	TS [13]	WM [14]	Crct. Rt%	Avg Tokens	NDVI ϵ_{vi}%	Ref B2 ϵ_{b2}%	AOD550 ϵ_{aod}%	LST ϵ_{lst}%	Built-S ϵ_{srf}%	Popul. ϵ_{pop}%	Loss ϵ_{ls}%	Canopy ϵ_{cvr}%	LCC acc%	Det $F1$%
GeoLLM-Eng.$^+$ [5]	–	–	39.84	19.09	5.37	8.38	6.28	6.61	7.65	6.62	9.74	6.66	70.13	66.09
	✓	–	41.86	20.09	7.74	12.49	8.62	9.43	10.70	10.73	9.00	7.20	86.25	70.53
	✓	✓	43.32	22.16	5.35	8.52	4.06	5.27	4.36	5.55	5.90	3.89	81.88	64.16
Chameleon [10]	–	–	39.26	23.80	**4.77**	7.87	4.58	5.07	4.97	5.05	4.98	3.69	30.52	49.34
	✓	–	40.14	21.49	4.87	8.13	4.32	4.78	4.24	5.03	5.21	4.03	33.89	57.97
	✓	✓	41.03	58.40	4.66	7.66	**4.02**	5.00	4.98	**4.87**	5.14	4.08	**85.82**	61.45
Magentic [8]	–	-	30.08	141.72	5.79	7.69	5.56	5.89	4.87	6.26	6.21	4.62	71.14	69.97
	✓	–	30.37	188.07	5.31	7.87	5.13	5.33	**4.24**	5.80	6.48	4.30	54.18	73.20
	✓	✓	33.98	210.11	7.67	10.15	6.36	7.62	8.05	7.22	7.67	5.41	73.08	76.59
GeoLLM-Squad	✓	✓	**60.29**	78.49	**4.77**	7.64	4.56	**4.14**	4.37	5.45	**4.58**	**3.61**	81.37	**78.58**

GeoLLM-Squad - Orchestrator prompting

≫ **Task**: You are a geospatial orchestrator [..]:
1. **Decompose** user request into agent subtasks.
2. **Generate** clear prompts for agents.
3. **Assign** agent order (e.g., load prior to filter).

≫ **Example**: "[..] 2024 NDVI data for Brisbane to [..] recommend crop rotation areas on the map."
≫ **Answer**: schedule = [Database (Load NDVI..), DataOps (Filter Brisbane), Agriculture (Recommend crop rotation areas based on ..), Map (Plot..)]

GeoLLM-Squad - Agents prompting

≫ **Task**: You are a geospatial expert [..]. Solve [..]
≫ **Query**: "[..] 2024 NDVI data for Brisbane to [..] recommend crop rotation areas on the map."
≫ **TS**: *Similar prompt*: "crop areas for Syndey."
Tools used: low_ndvi_clusters() , [..]
≫ **WM**: *Similar workflow*: "NDVI [..] for Melbourne."
Agents involved: Database , DataOps , [..]

To enhance agent prompting, we leverage two state-of-the-art practices, namely intent-based tool selection **TS** [13, 11] and agent workflow "memory" **WM** [14, 54]. **TS** provides tool recommendations via similarity search against a pre-compiled "training set" of prompt-solution pairs, providing *intra-agent* few-shot guidance. Orthogonally, **WM** compiles prompt-solution pairs at the workflow level, offering *inter-agent* few-shot guidance. This improves agent reasoning and prompting. For instance, if the agriculture agent fails its task due to missing NDVI data, its response flags the dependency, prompting the Orchestrator to involve the database agent.

IV. RESULTS

Metrics. We follow established evaluation practices and we report agentic correctness and costs [8, 13]. Drawing from [12, 55], we compute *correctness rate* as the ratio of correct tool-calling steps, i.e., how likely it is for the agent to invoke the correct functions in the expected order. To capture the impact of incorrect agentic actions to RS-specific metrics, we start from the fact that each task operates over sets of geospatial datapoints (e.g., NDVI values around location, or LST values for a date range). For each prompt, we record the data accessed by the agent and compare against the "gold" ground-truth set. For each missing item, we count its value as error and we compute mean-square percentage error ϵ following [56] across all 2k prompts. This assessment has been shown to better capture impact on downstream tasks instead of generic success rates or text-similary scores [3, 5].

Baselines. Table III provides a comprehensive evaluation of agentic performance and metrics across different RS tasks with GPT-4o-mini [50]. We compare GeoLLM-Squad against state-of-the-art methods, including the single-agent GeoLLM-Engine [5], as well as the multi-agent Chameleon [10] and Magentic [8] frameworks. Note that, since GeoLLM-Engine fails to scale past 3 combined tasks (discussed next), we evaluate it on each individual task in isolation.

GeoLLM-Engine correctness ranges from 39.84% to 43.32% for an average cost of 20.45k tokens, with the inclusion of tool selection (TS) and workflow memory (WM) improving performance. Magentic exhibits substantial communication overhead, with token costs exceeding 200k while achieving a correctness rate of only 33.98%. Although it achieves competitive performance for some metrics, such as detection F1 score (76.59%), the approach is inefficient due to excessive retry loops and repeated plan updates. Chameleon achieves efficient composition-based orchestration, reducing average token usage to about 40.14k with correctness rates comparable to the single-agent baseline.

GeoLLM-Squad. By separating agentic orchestration from geospatial task-solving, GeoLLM-Squad significantly outperforms all baselines with a correctness rate of 60.29%, reflecting a 17% improvement compared to GeoLLM-Engine, while maintaining competitive token cost of 78.49k per task. Overall, we achieve the lowest error rates across most domains, including NDVI ($\epsilon_{vi} = 4.77\%$), LST ($\epsilon_{lst} = 4.14\%$), and tree-loss and coverage ($\epsilon_{ls} = 4.58\%$, $\epsilon_{cvr} = 3.61\%$), showcasing the ability to balance accuracy and agentic efficiency.

Scaling across domains. We evaluate each method separately for each task (denoted as $^+$). As shown in Figure 2, the per-task correctness across the considered baselines

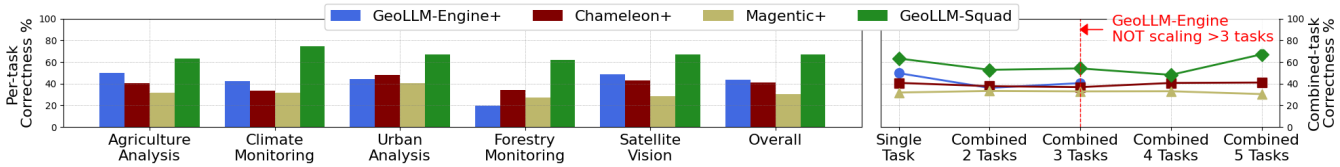

Fig. 2. Single- *vs*. multi-agent scalability ablations: we evaluate each method (with TS [13] and WM [14] incorporated) separately for each task and for multi-task combinations. GeoLLM-Squad consistently provides better agentic performance across all tasks.

shows Chameleon outperforming GeoEngine-LLM in some tasks, while the single-agent performs better in others. Overall, GeoLLM-Squad consistently provides better correctness across all tasks. Next, we ablate the scalability limitations and we evaluate multi-task combinations for each method. As the number of combined tasks increases, global prompt complexity and token requirements grow significantly, leading to context-window constraints that cause single-agent systems to *fail beyond three domains* (approximately 300 tools). In contrast, multi-agent setups like GeoLLM-Squad effectively distribute the toolset across specialized agents, maintaining robust performance as task complexity scales.

Scaling across LLM families. The performance of multi-agent systems depends on effective orchestration, a task where smaller language models (SLMs) typically struggle [7]. For example, the Magentic framework explicitly emphasizes the necessity of GPT-based models for orchestration [8]. However, this requirement imposes significant limitations for geoscientists due to cloud AI costs. Therefore, a critical property of effective multi-agent systems is their ability to perform well with open-source SLMs. To evaluate this, we test performance with Qwen-2.5 models (3B and 7B parameters) [57] running on A100 GPUs (Table IV). Chameleon's correctness rate drops by up to 20% when replacing GPT-4o-mini with Qwen-2.5-3B, while Magentic collapses entirely to less than 9% (effective "noise" levels) due to its dependence on advanced GPT models for its complex ledger-based logic. In contrast, GeoLLM-Squad demonstrates robust scalability across both SLMs, with the 7B variant achieving correctness rates comparable to the GPT-driven Chameleon. Of course, and despite GeoLLM-Squad achieving the highest performance among all methods, we emphasize the challenges faced by SLMs in performing agentic functions, as documented by practitioners in other domains [58]. We hope our findings motivate further *advancements in open-source geospatial SLMs* to bridge this gap and enhance their applicability in multi-agent RS systems.

TABLE IV
COMPARISON ACROSS GPT-4O-MINI AND QWEN-2.5 [57] (7B, 3B) MODELS. ALL METHODS INCORPORATE TS [13] AND WM [14]

LLM	Method	Crct. Rt %	Avg ϵ%
Qwen-2.5-7B	Chameleon [10]	25.54	6.49
	Magentic [8]	9.29	–
	GeoLLM-Squad	**40.29**	**6.27**
Qwen-2.5-3B	Chameleon [10]	21.11	**6.52**
	Magentic [8]	7.81	–
	GeoLLM-Squad	**36.95**	7.31

V. FUTURE WORK AND OPPORTUNITIES

As the first work to apply multi-agency to RS workflows, to our knowledge, our findings highlight several opportunities for future research. First, as discussed, improving the agentic performance of SLMs remains an urgent challenge. This could involve injecting domain-specific RS knowledge into SLMs prompting workflows [6] to compensate for the less powerful reasoning capabilities of SLMs. For the RS community, this is particularly important to ensure that tasks such as climate analysis and agricultural monitoring can be pursued with sustainable AI solutions [59], avoiding the prohibitive costs associated with multi-billion-parameter LLMs.

Second, while GeoLLM-Squad evaluates performance on multi-task benchmarks, these tasks primarily involve functional dependencies (e.g., simultaneously loading climate and agriculture data) rather than deeper semantic interdependencies. Future benchmarks should develop multi-agency workflows at a level beyond task delegation that require meaningful cross-domain reasoning [60], such as enabling maritime analysts to correlate coastal drought conditions derived from climate data with spikes in illegal fishing activities [61]. Current geospatial copilots lack the capability to handle this nuanced level of cross-agent synergy, presenting an exciting opportunity for the RS community.

VI. CONCLUSION

Single-agent geospatial methods face scalability limitations when applied across multiple RS domains due to finite LLM context windows. In this work, we introduced GeoLLM-Squad, a multi-agent system that *separates agentic orchestration and geospatial task-solving* processes which, combined with tool and workflow memory, improves both correctness and scalability across diverse RS applications. GeoLLM-Squad outperformed state-of-the-art baselines by up to 17% in task correctness while maintaining competitive model efficiency.

ACKNOWLEDGEMENTS

This work is partially supported by NSF CCF 2324854 and NSF-22-527. All open-source data products and frameworks used in this work, within non-commercial contexts and solely for research purposes, are fully cited in accordance with NASA ESDIS's open data policy towards reproducibility. The views and conclusions expressed herein are those of the authors and do not represent official policies or endorsements of Microsoft, funding agencies, or the U.S. government.

References

[1] K. Kuckreja, M. S. Danish, M. Naseer, A. Das, S. Khan, and F. S. Khan, "Geochat: Grounded large vision-language model for remote sensing," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2024, pp. 27 831–27 840.

[2] Y. Zhan, Z. Xiong, and Y. Yuan, "Skyeyegpt: Unifying remote sensing vision-language tasks via instruction tuning with large language model," *arXiv preprint arXiv:2401.09712*, 2024.

[3] S. Singh, M. Fore, and D. Stamoulis, "Evaluating tool-augmented agents in remote sensing platforms," *arXiv preprint arXiv:2405.00709*, 2024.

[4] H. Guo, X. Su, C. Wu, B. Du, L. Zhang, and D. Li, "Remote sensing chatgpt: Solving remote sensing tasks with chatgpt and visual models," in *IGARSS 2024 - 2024 IEEE International Geoscience and Remote Sensing Symposium*, 2024, pp. 11 474–11 478.

[5] S. Singh, M. Fore, and D. Stamoulis, "Geollm-engine: A realistic environment for building geospatial copilots," in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) Workshops*, June 2024, pp. 585–594.

[6] D. Stamoulis and D. Marculescu, "Geo-olm: Enabling sustainable earth observation studies with cost-efficient open language models & state-driven workflows," *arXiv preprint arXiv:2504.04319*, 2025.

[7] S. Hong, X. Zheng, J. Chen, Y. Cheng, J. Wang, C. Zhang, Z. Wang, S. K. S. Yau, Z. Lin, L. Zhou *et al.*, "Metagpt: Meta programming for multi-agent collaborative framework," *arXiv preprint arXiv:2308.00352*, 2023.

[8] A. Fourney, G. Bansal, H. Mozannar, C. Tan, E. Salinas, F. Niedtner, G. Proebsting, G. Bassman, J. Gerrits, J. Alber *et al.*, "Magentic-one: A generalist multi-agent system for solving complex tasks," *arXiv preprint arXiv:2411.04468*, 2024.

[9] Q. Wu, G. Bansal, J. Zhang, Y. Wu, S. Zhang, E. Zhu, B. Li, L. Jiang, X. Zhang, and C. Wang, "Autogen: Enabling next-gen llm applications via multi-agent conversation framework," *arXiv preprint arXiv:2308.08155*, 2023.

[10] P. Lu, B. Peng, H. Cheng, M. Galley, K.-W. Chang, Y. N. Wu, S.-C. Zhu, and J. Gao, "Chameleon: Plug-and-play compositional reasoning with large language models," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[11] M. Fore, S. Singh, and D. Stamoulis, "Geckopt: Llm system efficiency via intent-based tool selection," in *Proceedings of the Great Lakes Symposium on VLSI 2024*, 2024, pp. 353–354.

[12] S. Zhou, F. F. Xu, H. Zhu, X. Zhou, R. Lo, A. Sridhar, X. Cheng, T. Ou, Y. Bisk, D. Fried, U. Alon, and G. Neubig, "Webarena: A realistic web environment for building autonomous agents," in *The Twelfth International Conference on Learning Representations*, 2024.

[13] V. Paramanayakam, A. Karatzas, I. Anagnostopoulos, and D. Stamoulis, "Less is more: Optimizing function calling for llm execution on edge devices," *arXiv preprint arXiv:2411.15399*, 2024.

[14] Z. Z. Wang, J. Mao, D. Fried, and G. Neubig, "Agent workflow memory," *arXiv preprint arXiv:2409.07429*, 2024.

[15] P. Dias, A. Tsaris, J. Bowman, A. Potnis, J. Arndt, H. L. Yang, and D. Lunga, "Oreole-fm: successes and challenges toward billion-parameter foundation models for high-resolution satellite imagery," in *Proceedings of the 32nd ACM International Conference on Advances in Geographic Information Systems*, 2024, pp. 597–600.

[16] U. Mall, C. P. Phoo, M. K. Liu, C. Vondrick, B. Hariharan, and K. Bala, "Remote sensing vision-language foundation models without annotations via ground remote alignment," in *The Twelfth International Conference on Learning Representations*, 2024.

[17] Y. Hu, J. Yuan, C. Wen, X. Lu, and X. Li, "Rsgpt: A remote sensing vision language model and benchmark," *arXiv preprint arXiv:2307.15266*, 2023.

[18] J. D. Silva, J. Magalhães, D. Tuia, and B. Martins, "Large language models for captioning and retrieving remote sensing images," *arXiv preprint arXiv:2402.06475*, 2024.

[19] F. Liu, D. Chen, Z. Guan, X. Zhou, J. Zhu, Q. Ye, L. Fu, and J. Zhou, "Remoteclip: A vision language foundation model for remote sensing," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–16, 2024.

[20] W. Zhang, M. Cai, T. Zhang, Y. Zhuang, and X. Mao, "Earthgpt: A universal multimodal large language model for multisensor image comprehension in remote sensing domain," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 62, pp. 1–20, 2024.

[21] Y. Jian, F. Yu, S. Singh, and D. Stamoulis, "Stable diffusion for aerial object detection," *arXiv preprint arXiv:2311.12345*, 2023.

[22] P. Bhandari, A. Anastasopoulos, and D. Pfoser, "Urban mobility assessment using llms," in *Proceedings of the 32nd ACM International Conference on Advances in Geographic Information Systems*, 2024, pp. 67–79.

[23] C. Yu, X. Xie, Y. Huang, and C. Qiu, "Harnessing llms for cross-city od flow prediction," in *Proceedings of the 32nd ACM International Conference on Advances in Geographic Information Systems*, 2024, pp. 384–395.

[24] C. Zhang, A. Sriram, K.-H. Hung, R. Wang, and D. Yankov, "Context-aware conversational map search with llm," in *Proceedings of the 32nd ACM International Conference on Advances in Geographic Information Systems*, 2024, pp. 485–488.

[25] J. Li, K. Lammers, X. Yin, X. Yin, L. He, R. Lu, and Z. Li, "Metafruit meets foundation models: Leveraging a comprehensive multi-fruit dataset for advancing agricultural foundation models," *arXiv preprint arXiv:2407.04711*, 2024.

[26] G. Yang, Y. Li, Y. He, Z. Zhou, L. Ye, H. Fang, Y. Luo, and X. Feng, "Multimodal large language model for wheat breeding: a new exploration of smart breeding," *arXiv preprint arXiv:2411.15203*, 2024.

[27] Microsoft Industry Clouds, "Evolving microsoft azure data manager for agriculture to transform data into intuitive insights," https://azure.microsoft.com/en-us/blog/evolving-microsoft-azure-data-manager-for-agriculture, November 2023, accessed: 2025-01-07.

[28] N. I. Bountos, A. Ouaknine, and D. Rolnick, "Fomo-bench: a multi-modal, multi-scale and multi-task forest monitoring benchmark for remote sensing foundation models," *arXiv preprint arXiv:2312.10114*, 2023.

[29] A. Lacoste, N. Lehmann, P. Rodriguez, E. Sherwin, H. Kerner, B. Lütjens, J. Irvin, D. Dao, H. Alemohammad, A. Drouin *et al.*, "Geo-bench: Toward foundation models for earth monitoring," *Advances in Neural Information Processing Systems*, vol. 36, 2024.

[30] X. X. Zhu, Z. Xiong, Y. Wang, A. J. Stewart, K. Heidler, Y. Wang, Z. Yuan, T. Dujardin, Q. Xu, and Y. Shi, "On the foundations of earth and climate foundation models," *arXiv preprint arXiv:2405.04285*, 2024.

[31] Z. Xiong, Y. Wang, F. Zhang, A. J. Stewart, J. Hanna, D. Borth, I. Papoutsis, B. Le Saux, G. Camps-Valls, and X. X. Zhu, "Neural plasticity-inspired foundation model for observing the earth crossing modalities," *arXiv e-prints*, pp. arXiv–2403, 2024.

[32] D. Kononykhin, M. Mozikov, K. Mishtal, P. Kuznetsov, D. Abramov, N. Sotiriadi, Y. Maximov, A. V. Savchenko, and I. Makarov, "From data to decisions: Streamlining geospatial operations with multimodal globeflowgpt," in *Proceedings of the 32nd ACM International Conference on Advances in Geographic Information Systems*. New York, NY, USA: Association for Computing Machinery, 2024, p. 649–652.

[33] Y. Chen, W. Wang, S. Lobry, and C. Kurtz, "An llm agent for automatic geospatial data analysis," *arXiv preprint arXiv:2410.18792*, 2024.

[34] V. G. Goecks and N. R. Waytowich, "Disasterresponsegpt: Large language models for accelerated plan of action development in disaster response scenarios," *arXiv preprint arXiv:2306.17271*, 2023.

[35] K. Didan, "Mod13a3 modis/terra vegetation indices monthly l3 global 1km sin grid v006," 2015, accessed 2025-01-07. [Online]. Available: https://doi.org/10.5067/MODIS/MOD13A3.006

[36] E. Vermote, C. Justice, M. Claverie, and B. Franch, "Mod09ga modis/terra surface reflectance daily l2g global 1km sin grid v006," 2015, accessed 2025-01-07. [Online]. Available: https://doi.org/10.5067/MODIS/MOD09GA.006

[37] Z. Wan, S. Hook, and G. Hulley, "Modis/terra land surface temperature/emissivity 8-day l3 global 1km sin grid v061," https://doi.org/10.5067/MODIS/MYD11A2.061, 2021, accessed 2025-01-07.

[38] A. Lyapustin and Y. Wang, "MCD19A2 MODIS/Terra+Aqua Land Aerosol Optical Depth Daily L2G Global 1km SIN Grid V006," https://doi.org/10.5067/MODIS/MCD19A2.006, 2018, accessed 2025-01-07.

[39] E. Commission and J. R. Centre, *GHSL data package 2023*. Publications Office of the European Union, 2023.

[40] M. Pesaresi and P. Politis, *GHS-BUILT-S R2023A - GHS built-up surface grid, derived from Sentinel2 composite and Landsat, multitemporal (1975-2030)*. European Commission, Joint Research Centre (JRC), 2023. [Online]. Available: http://data.europa.eu/89h/9f06f36f-4b11-47ec-abb0-4f8b7b1d72ea

[41] M. C. Hansen, P. V. Potapov, R. Moore, M. Hancher, S. A. Turubanova, A. Tyukavina, D. Thau, S. V. Stehman, S. J. Goetz, T. R. Loveland *et al.*, "High-resolution global maps of 21st-century forest cover change," *science*, vol. 342, no. 6160, pp. 850–853, 2013.

[42] D. Lam, R. Kuzma, K. McGee, S. Dooley, M. Laielli, M. Klaric, Y. Bulatov, and B. McCord, "xview: Objects in context in overhead imagery," *arXiv preprint arXiv:1802.07856*, 2018.

[43] R. Gupta, B. Goodman, N. Patel, R. Hosfelt, S. Sajeev, E. Heim, J. Doshi, K. Lucas, H. Choset, and M. Gaston, "Creating xbd: A dataset for assessing building damage from satellite imagery," in *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition workshops*, 2019, pp. 10–17.

[44] F. Paolo, T.-t. T. Lin, R. Gupta, B. Goodman, N. Patel, D. Kuster, D. Kroodsma, and J. Dunnmon, "xview3-sar: Detecting dark fishing activity using synthetic aperture radar imagery," *Advances in Neural Information Processing Systems*, vol. 35, pp. 37 604–37 616, 2022.

[45] C. Luckett, B. McCarthy, T.-T. Cao, and A. Robles-Kelly, "The sarfish dataset and challenge," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, 2024, pp. 752–761.

[46] X. Sun, P. Wang, Z. Yan, F. Xu, R. Wang, W. Diao, J. Chen, J. Li, Y. Feng, T. Xu *et al.*, "Fair1m: A benchmark dataset for fine-grained object recognition in high-resolution remote sensing imagery," *ISPRS Journal of Photogrammetry and Remote Sensing*, vol. 184, pp. 116–130, 2022.

[47] G. Christie, N. Fendley, J. Wilson, and R. Mukherjee, "Functional map of the world," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 6172–6180.

[48] G. Sumbul, M. Charfuelan, B. Demir, and V. Markl, "Bigearthnet: A large-scale benchmark archive for remote sensing image understanding," in *IGARSS 2019-2019 IEEE International Geoscience and Remote Sensing Symposium*. IEEE, 2019, pp. 5901–5904.

[49] Earth Science Data and Information System (ESDIS), "NASA Earthdata Search," https://search.earthdata.nasa.gov, 2025, accessed 2025-01-07.

[50] OpenAI, "Gpt-4o-mini: Advancing cost-efficient intelligence," https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/, 2025, accessed: January 15, 2025.

[51] A. Hurst, A. Lerer, A. P. Goucher, A. Perelman, A. Ramesh, A. Clark, A. Ostrow, A. Welihinda, A. Hayes, A. Radford *et al.*, "Gpt-4o system card," *arXiv preprint arXiv:2410.21276*, 2024.

[52] Y. Zhuang, Y. Yu, K. Wang, H. Sun, and C. Zhang, "Toolqa: A dataset for llm question answering with external tools," *Advances in Neural Information Processing Systems*, vol. 36, pp. 50 117–50 143, 2023.

[53] A. Narayan, M. F. Chen, K. Bhatia, and C. Ré, "Cookbook: A framework for improving llm generative abilities via programmatic data generating templates," *arXiv preprint arXiv:2410.05224*, 2024.

[54] V. K. Srinivasan, Z. Dong, B. Zhu, B. Yu, D. Mosk-Aoyama, K. Keutzer, J. Jiao, and J. Zhang, "Nexusraven: a commercially-permissive language model for function calling," in *NeurIPS 2023 Foundation Models for Decision Making Workshop*, 2023.

[55] J. Y. Koh, R. Lo, L. Jang, V. Duvvur, M. C. Lim, P.-Y. Huang, G. Neubig, S. Zhou, R. Salakhutdinov, and D. Fried, "Visualwebarena: Evaluating multimodal agents on realistic visual web tasks," in *ICLR 2024 Workshop on Large Language Model (LLM) Agents*, 2024.

[56] E. Cai, D.-C. Juan, D. Stamoulis, and D. Marculescu, "Neuralpower: Predict and deploy energy-efficient convolutional neural networks," in *Asian Conference on Machine Learning*. PMLR, 2017, pp. 622–637.

[57] A. Yang, B. Yang, B. Zhang, B. Hui, B. Zheng, B. Yu, C. Li, D. Liu, F. Huang, H. Wei *et al.*, "Qwen2. 5 technical report," *arXiv preprint arXiv:2412.15115*, 2024.

[58] vLLM Team, "vllm documentation," https://docs.vllm.ai/en/stable/, 2025, accessed: January 15, 2025.

[59] R. Roscher, M. Rußwurm, C. Gevaert, M. Kampffmeyer, J. A. dos Santos, M. Vakalopoulou, R. Hänsch, S. Hansen, K. Nogueira, J. Prexl *et al.*, "Data-centric machine learning for geospatial remote sensing data," *arXiv preprint arXiv:2312.05327*, 2023.

[60] I. Tsoumas, G. Giannarakis, V. Sitokonstantinou, A. Koukos, D. Loka, N. Bartsotas, C. Kontoes, and I. Athanasiadis, "Evaluating digital tools for sustainable agriculture using causal inference," *arXiv preprint arXiv:2211.03195*, 2022.

[61] D. Kroodsma, J. Turner, C. Luck, T. Hochberg, N. Miller, P. Augustyn, and S. Prince, "Global prevalence of setting longlines at dawn highlights bycatch risk for threatened albatross," *Biological Conservation*, vol. 283, p. 110026, 2023.