VisKnacks: Agentic ParaView Visualization on HPC

Nicholas M. Synovic
George K. Thiruvathukal
Department of Computer Science
Loyola University Chicago
Chicago, IL, USA
{nsynovic,gkt}@luc.edu

Manish Mathai
University of Oregon
Eugene, OR, USA
mmathai@uoregon.edu

Mengjiao Han, Joseph Insley,
Janet Knowles, Victor Mateevitsi,
Silvio Rizzi, and Michael E. Papka
Leadership Computing Facility
Argonne National Laboratory
Lemont, IL, USA
{hanm,insley,jknowles,
vmateevitsi,srizzi,papka}@anl.gov

***Abstract*—Artificial intelligence can assist in prototyping scientific visualizations. To assist visualization experts, we introduce *VisKnacks*, an autonomous ParaView visualization agent enabled by large language models (LLMs). VisKnacks integrates multi-agent orchestration, retrieval-augmented generation, and containerized rendering services for high-performance computing (HPC) systems while remaining compatible with existing agent harnesses. The system supports hybrid workflows in which ParaView rendering is distributed to HPC resources while pipeline creation and iterative refinement remain local. We present the system overview, initial benchmarks visualization quality benchmarks, and lessons learned from deploying this system.**

***Index Terms*—Scientific visualization, high-performance computing, visualization agents**

I. Introduction

Scientific simulations on HPC systems can produce terabytes of data per timestep [1], requiring significant expert effort to process and visualize. LLM enabled scientific visualization agents (*i.e.,* SciVisAgents) have been proposed to support dataset exploration and visualization prototyping from natural language prompts [2, 3]. Prior work has proposed several systems [4–12], to address this challenge. However, anecdotes from our visualization colleagues suggest that existing SciVisAgents often target local execution and provide limited support for HPC capabilities (*e.g.,* remote rendering, deployed visualization-service integration, software packaging) limiting exploration of large datasets.

We introduce *VisKnacks*, a SciVisAgent for hybrid local—HPC visualization workflows. VisKnacks packages agent accessible visualization services for HPC deployment and provides plugable components for existing agent harnesses. It enables agents to develop ParaView [13] pipelines locally, render remotely on HPC resources, and use returned logs and images to refine those pipelines. We evaluate VisKnacks on ParaView tasks from SciVisAgentBench [14], using Argonne National Laboratory's Leadership Computing Facility Crux HPC system.

II. System Overview

VisKnacks includes: a multi-agent prompt interpreter for converting user requests into structured visualization queries, an Agent Skill [15] for common ParaView pipeline operations, and two MCP services for ParaView Python API documentation and rendering pipelines on HPC systems. These components integrate with existing agent harnesses (*e.g.,* OpenCode [16]).

TABLE I
Visualization quality on 10 SciVisAgentBench ParaView tasks. Higher scores are better.

Model	Mean PSNR ↑	Mean SSIM ↑
VisKnacks + Claude	20.72	0.80
VisKnacks + Gemini	17.62	0.77
VisKnacks + GPT	15.27	0.73

III. Initial Results

We evaluate VisKnacks on 10 ParaView tasks from SciVisAgentBench. For each task, VisKnacks generates and executes a ParaView pipeline on ALCF Crux using Anthropic Claude Opus 4.8 (Claude) [17], Google Gemini 3.5 Flash (Gemini) [18], and OpenAI GPT-5.5 (GPT) [19]. Each model—task combination was evaluated using the same task prompts, tools, and configuration. We compare each rendered image to the human generated task reference image on PSNR and SSIM as reproducible measures of image visual quality.

Table I reports each model's mean score across tasks.

IV. Lessons Learned

SciVisAgents can reduce the expert effort required to prototype visualization pipelines by autonomously generating and refining candidate workflows. VisKnacks demonstrates agent driven visualization can distinguish interactive development from data/compute-intensive execution. ParaView pipelines can be refined locally while HPC resources render large resident datasets.

VisKnacks demonstrates a portable path for deploying SciVisAgents. By using existing interfaces and packaging mechanisms, the system makes documentation retrieval, pipeline generation, and remote rendering available to existing agent harnesses. This modular design supports the integration and evaluation of different SciVisAgent components

Deploying VisKnacks on ALCF Crux revealed challenges in data exploration and system configuration. Because full simulation datasets are impractical to provide to LLMs due to their

large volumes of data, VisKnacks uses file metadata to infer pipeline components such as loaders and data dimensionality thereby requiring upfront exploration.

V. Reproducibility

VisKnacks source code is available at this GitHub URL. VisKnacks packages its MCP services as Apptainer [20] images, distributes its Agent Skill through `skills.sh` [21], and provides preconfigured subagent definitions for supported agent harnesses.

VI. Conclusions and Next Steps

We presented *VisKnacks*, an open source, LLM agnostic SciVisAgent system for hybrid local–HPC ParaView workflows. VisKnacks combines an agent harness, multi-agent prompt interpretation, an Agent Skill, and MCP services for ParaView documentation retrieval and remote rendering. By packaging HPC-accessible rendering services and integrating them with existing agent harness interfaces, VisKnacks enables agents to iteratively develop pipelines locally and render them on remote HPC resources.

We evaluated three LLMs on SciVisAgentBench ParaView tasks using PSNR and SSIM between HPC rendered and reference images.

Future work will evaluate VisKnacks with additional visualization backends and datasets, extend it to in-situ visualization, and investigate workflow-management frameworks for scheduler-compatible deployment of remote rendering services More broadly, we will assess whether a shared, model agnostic harness can improve the portability, reproducibility, and evaluation of SciVisAgent workflows.

Acknowledgments

This research used resources of the Argonne Leadership Computing Facility, which is a U.S. Department of Energy Office of Science User Facility operated under contract DE-AC02-06CH11357. This work was supported by the U.S. Department of Energy, Office of Science (SC), Advanced Scientific Computing Research (ASCR), Competitive Portfolios Project, PIONEER: Program for Intelligent Optimization for Next-generation Experiments, Exploration, and Research, under Contract DE-AC02-06CH11357.

References

[1] Y. Ye, F. Sauer, K.-L. Ma, K. Aditya, and J. Chen, "A user-centered design study in scientific visualization targeting domain experts," vol. 26, no. 6, pp. 2192–2203. [Online]. Available: https://ieeexplore.ieee.org/document/8977377

[2] K. Ai, H. Miao, Z. Li, C. Wang, and S. Liu, "An evaluation-centric paradigm for scientific visualization agents," version Number: 1. [Online]. Available: https://arxiv.org/abs/2509.15160

[3] V. Dhanoa, A. Wolter, G. M. León, H.-J. Schulz, and N. Elmqvist, "Agentic visualization: Extracting agent-based design patterns from visualization systems," vol. 45, no. 6, pp. 89–100. [Online]. Available: https://ieeexplore.ieee.org/document/11153807

[4] A. Biswas, T. L. Turton, N. R. Ranasinghe, S. Jones, B. Love, W. Jones, A. Hagberg, H.-W. Shen, N. DeBardeleben, and E. Lawrence, "VizGenie: Toward self-refining, domain-aware workflows for next-generation scientific visualization," vol. 32, no. 1, pp. 1021–1031. [Online]. Available: https://ieeexplore.ieee.org/document/11279991

[5] D. Jia, A. Irger, L. Besançon, O. Strnad, D. Luo, J. Björklund, A. Kouyoumdjian, A. Ynnerman, and I. Viola, "VOICE: Visual oracle for interaction, conversation, and explanation," vol. 31, no. 10, pp. 8828–8845. [Online]. Available: https://ieeexplore.ieee.org/document/11037292

[6] S. Liu, H. Miao, Z. Li, M. Olson, V. Pascucci, and P.-T. Bremer, "AVA: Towards autonomous visualization agents through visual perception-driven decision-making," vol. 43, no. 3, p. e15093, _eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/cgf.15093. [Online]. Available: https://onlinelibrary.wiley.com/doi/abs/10.1111/cgf.15093

[7] R. Massa, A. Shaik, B. Ta, M. Han, J. Insley, J. Knowles, V. A. Mateevitsi, M. E. Papka, S. Rizzi, and S. Shilpika, "GENIUS: AI powered assistant for scientific research," in *2025 IEEE International Conference on eScience (eScience)*. IEEE, pp. 347–348. [Online]. Available: https://ieeexplore.ieee.org/document/11181556/

[8] T. Mallick, O. Yildiz, D. Lenz, and T. Peterka, "ChatVis: Automating scientific visualization with a large language model," in *SC24-W: Workshops of the International Conference for High Performance Computing, Networking, Storage and Analysis*. IEEE, pp. 49–55. [Online]. Available: https://ieeexplore.ieee.org/document/10820680/

[9] Model Context Protocol. Model context protocol specification. Version number: 2025-11-25. [Online]. Available: https://modelcontextprotocol.io/specification/2025-11-25

[10] S. Liu, H. Miao, and P.-T. Bremer, "ParaView-MCP: An autonomous visualization agent with direct tool use," in *2025 IEEE Visualization and Visual Analytics (VIS)*, pp. 61–65, ISSN: 2771-9553. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/11298840

[11] K. Ai, H. Miao, K. Tang, S. Liu, and C. Wang, "SciVisAgentSkills: Design and evaluation of agent skills for scientific data analysis and visualization." [Online]. Available: http://arxiv.org/abs/2606.05525

[12] T. Peterka, T. Mallick, O. Yildiz, D. Lenz, C. Quammen, and B. Geveci, "ChatVis: Large language model agent for generating scientific visualizations," in *2025 IEEE 15th Symposium on Large Data Analysis and Visualization (LDAV)*, pp. 22–32, ISSN: 2832-6512. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/11298870

[13] J. Ahrens, B. Geveci, and C. Law, "ParaView: An end-user tool for large-data visualization," in *Visualization*

handbook. Elsevier, pp. 717–731, tex.hal_id: hal-05568143 tex.hal_version: v1. [Online]. Available: https://hal.science/hal-05568143

[14] K. Ai, H. Miao, K. Tang, N. Gorski, J. Sun, G. Liu, H. I. Ingolfsson, D. Lenz, H. Guo, H. Yu, T. Leburu, M. Molash, B. Wang, T. Peterka, C. Wang, and S. Liu, "SciVisAgentBench: A benchmark for evaluating scientific data analysis and visualization agents," version Number: 1. [Online]. Available: https://arxiv.org/abs/2603.29139

[15] Agent Skills. Agent skills specification. [Online]. Available: https://agentskills.io/specification

[16] "anomalyco/opencode," original-date: 2025-04-30T20:08:00Z. [Online]. Available: https://github.com/anomalyco/opencode

[17] Anthropic. Introducing claude opus 4.8. [Online]. Available: https://www.anthropic.com/news/claude-opus-4-8

[18] Google DeepMind. Gemini 3.5 flash - model card. [Online]. Available: https://deepmind.google/models/model-cards/gemini-3-5-flash/

[19] OpenAI. GPT-5.5 system card. [Online]. Available: https://openai.com/index/gpt-5-5-system-card/

[20] G. M. Kurtzer, cclerget, M. Bauer, I. Kaneshiro, D. Trudgian, and D. Godlove, "hpcng/singularity: Singularity 3.7.3." [Online]. Available: https://zenodo.org/records/4667718

[21] "vercel-labs/skills," original-date: 2026-01-14T21:31:18Z. [Online]. Available: https://github.com/vercel-labs/skills