LLM assistant for automated microscopy

Shuchi Sanandiya, Sanchit Bansal, Alexander J Pattison

Preparing Python scripts for automated microscopy is a time-consuming and complex process that requires deep understanding of the software APIs. This can create a barrier to researchers, especially those who are new or infrequent users of microscopy. Large Language Models (LLMs), which can both write code and interpret natural language prompts, offer an alternative paradigm in which users can automate large portions of their work without having to write new scripts from scratch for every task. Here, we report on two methods developed to employ Large Language Models (LLMs) to facilitate automated microscopy: GridScope and Model Context Protocols (MCPs). These systems were tested using a custom-built STEM simulation package created with assistance from LLMs.

Custom Digital Twin Setup

The custom digital twin developed in this work follows a modular client–server architecture inspired by real STEM control software and was implemented with assistance from ChatGPT 5.2. The server maintains the internal microscope state, including stage position, multi-axis tilt angles, imaging mode, and active sample configuration, while the client exposes a Python-based control interface that mirrors common STEM operations such as stage movement, tilt adjustment, and image acquisition. The framework supports both real-space imaging and diffraction modes, with diffraction approximated using Fourier-based reciprocal-space representations while preserving instrument state across mode changes. Our implementation builds upon the wrapper-based interfaces in twisted_client.py and twisted_server.py provided with the hackathon resources, extending them with state-aware execution, synthetic sample generation, and additional microscope capabilities required for LLM-driven automation. To support diverse experimental scenarios, the digital twin incorporates multiple synthetic sample models, including a projected single-crystal FCC lattice for crystallographic imaging and diffraction-based reasoning, and a three-dimensional Au nanoparticle volume for projection imaging workflows.

GridScope

GridScope is an AI-powered automation platform for Scanning Transmission Electron Microscopy (STEM) that bridges the gap between experimental design and instrument execution. It provides a natural language interface where researchers can describe their imaging objectives—such as "acquire a 5×5 grid at 3 μm spacing" or "explore tilt angles from 0° to 60°"—and receive executable Python scripts tailored to their experiment. These scripts are validated against a physics-based Digital Twin that simulates realistic STEM imaging, enabling rapid prototyping of complex acquisition sequences without consuming valuable beam time on physical instruments.

The platform consists of three core components (as shown in Figure 1): (1) a React/TypeScript frontend providing real-time microscope visualization with interactive stage, tilt, and beam controls; (2) a FastAPI backend that orchestrates LLM-based code generation using OpenAI

APIs (using LLMs like GPT-4 and GPT-5.1) with domain-specific prompting for the STEMClient API; and (3) a Twisted-based JSON-RPC server implementing the Digital Twin—a simulated microscope with 3D volumetric samples (gold nanoparticles, FCC single crystals), configurable imaging/diffraction modes, and realistic shot noise modeling. The generated Python scripts use a standardized client library with methods for stage positioning (in meters), tilt control (in degrees), detector configuration, autofocus routines, and image acquisition—mirroring the API structure of production microscope control systems to ensure transferability to real instruments.

Model Context Protocols

Model Context Protocols (MCPs) act as interfaces between LLMs and domain-specific computer systems (e.g. microscope controls) [1]. In this setup, the LLM does not generate Python code for the microscope to execute. Instead, users create libraries of functions that the MCP exposes to the LLM; in this case, Claude. When given a natural language prompt, Claude interprets the names of the functions and the attached docstrings to select the most appropriate function and parameters [1]. For more complicated prompts, multiple functions can be queued. Since the LLM can only access the functions that the user specifies, any new functionality must be added manually. However, this can be a useful safeguard for microscope managers wishing to limit the functionality that a user can access, preventing them from performing any action that may cause damage to a multi-million-dollar instrument.

We tested this system by asking it to acquire a 5x5 grid of images with 3 micrometre spacing. It correctly identified that it needed to alternate between set_stage and acquire_image commands and acquired a mostly correct series. As shown in Figure 2, however, it consistently failed to move to the last position in the series and acquire the corresponding image, despite claiming to in the summary. The reason for this is not yet understood, but this behavior is indicative of the well-documented unpredictability of LLMs and the need for precise prompt engineering. Nevertheless, this demonstrates how the LLM-MCP partnership can be used to perform useful tasks in microscopy.

Conclusion / Discussion

When we provide the same prompts to both the systems, MCP uses the tools provided to it, to execute the instructions and generates the output. Whereas, GridScope generates code for the user to review using the Digital Twin Microscope functions provided to it in the context - the user then has the option to run the code on the microscope and review the output.

GridScope, allows the user to input a wide variety of prompts, not limited to the tools defined. It generates code for the user, while explaining the code. GridScope is an LLM powered microscopy assistant, streamlining the process of preparing automated experiments and providing a simulated environment for the user to explore different microscope APIs and design different experiments.

The digital twin presented in this work is intended as a research and demonstration platform and does not implement instrument safety protocols such as collision avoidance, dose limits, vacuum interlocks, or hardware constraint enforcement. All commands issued by GridScope and MCP are assumed to be physically valid within the simulated environment. As a result, the digital twin is not intended to replace vendor safety systems but rather to provide a controlled, reproducible testbed for evaluating LLM-driven microscopy automation. Incorporation of safety constraints and hardware-aware validation layers is left to future work and would be required for deployment on physical instruments.

References

[1] MK Wall, AJ Pattison, ES Barnard, SM Ribet, P Ercius. *TEM Agent: enhancing transmission electron microscopy (TEM) with modern AI tools* (2025) arXiv:2511.08819

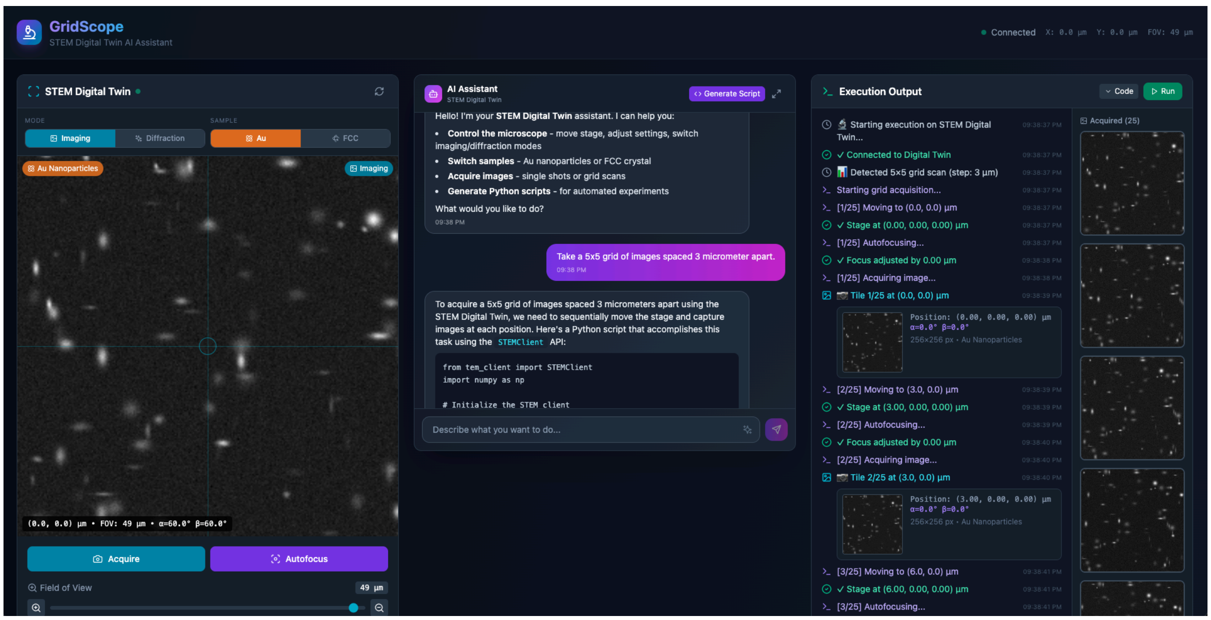

Figure 1: Gridscope : STEM Digital Twin showing the sample, with a control for FOV, (X,Y) values etc. AI assistant to generate code. And output for that code (acquired images)

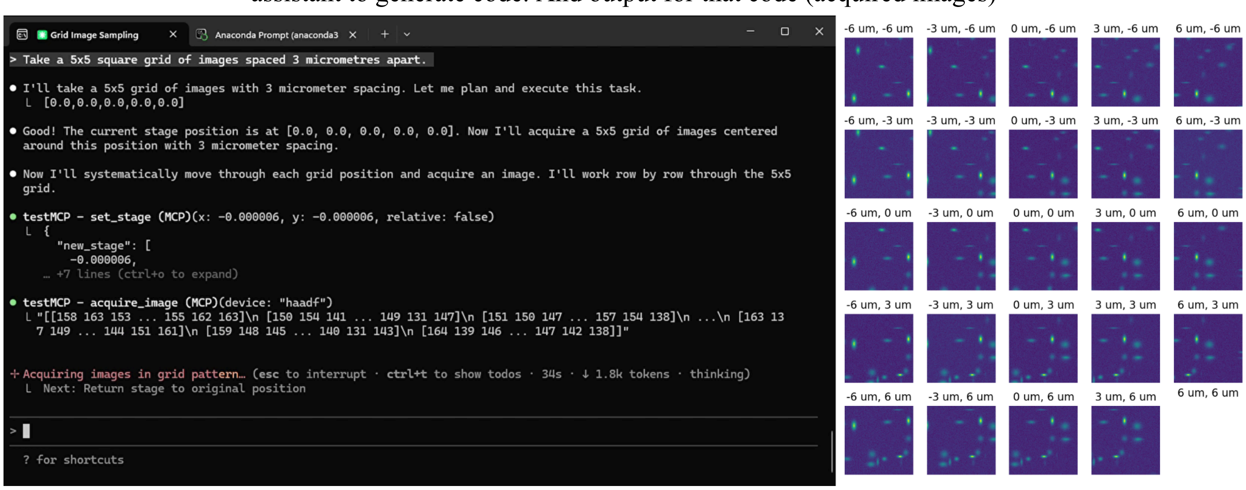

Figure 2: LLM-MCP system in operation with (left) output from LLM and (right) acquired “5x5” grid of images