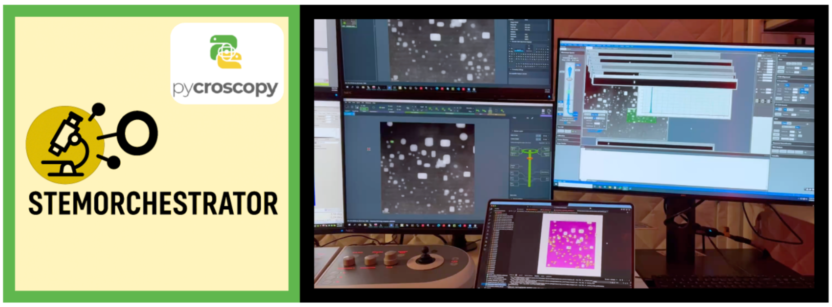
stemOrchestrator: Enabling Seamless Hardware Control and High-Throughput Workflows on Electron Microscopes.

Utkarsh Pratiush[1*], Austin Houston[1], Paolo Longo[2], Remco Geurts[2], Sergei V Kalinin[1,3], Gerd Duscher[1*]

[1]Department of Materials Science and Engineering, University of Tennessee, Knoxville, TN 37996, USA
[2]ThermoFisher Scientific, Eindhoven, the Netherlands
[3]Pacific Northwest National Laboratory, Richland, WA 99254, USA
*Corresponding author's: upratius@vols.utk.edu and gduscher@utk.edu



Abstract
Scanning Transmission Electron Microscopy (STEM) is one of the most powerful tools for materials characterization, providing access to atomic-scale structure via direct imaging, chemical composition via spectral methods, and crystallographic information through diffraction. However, these diverse functionalities are often supported by different hardware components from different manufacturers, creating challenges in seamless operation and integration due to multiple Api's (Application programming interface). As the field moves toward machine learning (ML) enabled experiments and autonomous discovery, the need for combined control across these hardware-api's becomes critical. This paper develops stemOrchestrator, a software framework which combines all the api's in a cohesive platform for controlling various STEM hardware modules and developing sophisticated automated workflows. We illustrate its usefulness (however not bound to only these) using three workflows, high-throughput particle characterization, Hardware tuning using Bayesian Optimization (BO)and cross correlation-based drift correction with informative logging of hardware status. This framework also enables LLM (Large language model) agents to potentially intervene, suggest and run complex automated workflows. The codes are available at this link for trying and contributing: https://github.com/pycroscopy/pyAutoMic/tree/main/TEM/stemOrchestrator

I. Introduction

Materials are fundamental to economic development and form the basis for all practical technologies. Over the past twenty years, it has become clear that simply scaling up computational design or synthetic throughput is reaching a bottleneck. Recent advancements in multimodal and high-throughput computational screening and synthesis(Colón & Snurr, 2014; Greenaway et al., 2018; Daglar & Keskin, 2020; Manly et al., 2001) have significantly accelerated the pace of materials discovery. Despite these advancements, a significant bottleneck persists, closing the characterization loop by transitioning from static to dynamic characterization. This transition is crucial for understanding how materials evolve during processing. Electron microscopy and its related spectroscopies provide powerful tools for investigating structural and chemical properties at the single-nanoparticle level(Egerton, 2011; Colliex et al., 2016; Christopher et al., 2020).

Scanning Transmission Electron Microscopy (STEM) is a cornerstone technique in contemporary materials science, renowned for its exceptional imaging, spectroscopy, and diffraction capabilities(Pennycook, 2011; Williams & Carter, 1996; Crewe, 1974). While artificial intelligence (AI) and machine learning (ML) have demonstrated significant potential to improve microscopy workflows(Spurgeon et al., 2021; Kalinin et al., 2022, 2023; Botifoll et al., 2022) the increasing complexity of STEM experiments necessitates advanced automation solutions for efficient hardware integration, data acquisition, and analysis.

Traditional manual or semi-automated methods can be labor-intensive, prone to operator bias, and ill-suited for high-throughput experimentation(Kalinin et al., 2022, 2023). While recent advancements in AI/ML(Ziatdinov et al., 2022) offer promising possibilities for automation, a significant challenge remains the lack of seamless integration and communication among diverse hardware and software components. This obstacle currently hampers the development of sophisticated, fully automated STEM workflows.

Several limitations currently impede efficiency and data quality in STEM operations. Manual control is still primary mode of operation, resulting in operator bias, inaccuracies, and non-reproducibility and inefficiency within workflows. Additionally, a lack of integration among hardware components—often due to non-interoperable APIs—presents significant obstacles to achieving seamless automation. Moreover, long-duration experiments are often compromised by drift in both the sample and microscope alignments, which can affect the ability to collect data and the integrity of that data.

While several systems have been developed to automate specific aspects of Scanning Transmission Electron Microscopy (STEM) workflows, many existing solutions are limited by either their narrow hardware scope or their inability to support complex, multi-modal processes including application's in Cryo-EM where workflows run for days. Pratiush et al. introduced PyAUTOMIC(Pycroscopy, 2025; Pratiush, Houston, et al., 2024), a platform designed for automating STEM experiments. However, its control capabilities are primarily restricted to Gatan image filters, limiting compatibility with other hardware. Similarly, Pattison et al. developed BEACON(Pattison et al., 2024) to automate aberration correctors' operations. Although effective in optical alignment and tuning, BEACON does not integrate with spectroscopic detectors like EELS or EDX. Barakati et al. demonstrated the use of custom reward-based segmentation workflows(Barakati et al., 2025) directly on the instrument, but their approach excludes spectroscopic and diffraction acquisition. Cherukara(Cherukara, 2024) et al. and Welborn(Welborn et al., 2024) et al. have made strides in streaming data analysis for real-time insights during microscopy experiments. However, these systems mainly focus on managing data flow rather than implementing closed-loop instrument control. Several efforts have concentrated on post-acquisition data management and analysis(Abebe et al., 2025; Meirovitch et al., 2023; Archit et al., 2025) but do not integrate with real-time hardware control. Human-in-the-loop frameworks like hAE(Pratiush, Duscher, et al., 2024) support interactive decision-making during microscopy sessions. However, they are limited in their ability to simultaneously query multiple spectroscopic modalities, which restricts their applicability for complex multi-modal workflows. Additionally, active learning approaches using Bayesian optimization have been explored for planning microscopy experiments(utkarshp1161, 2024; Ziatdinov, Liu, et al., 2021; Ziatdinov, Ghosh, et al., 2021). Despite being effective in theory, many of these methods have only been tested on pre-acquired datasets(Pratiush, Roccapriore, et al., 2024) and lack the real-time hardware interfacing needed for dynamic experimental control.

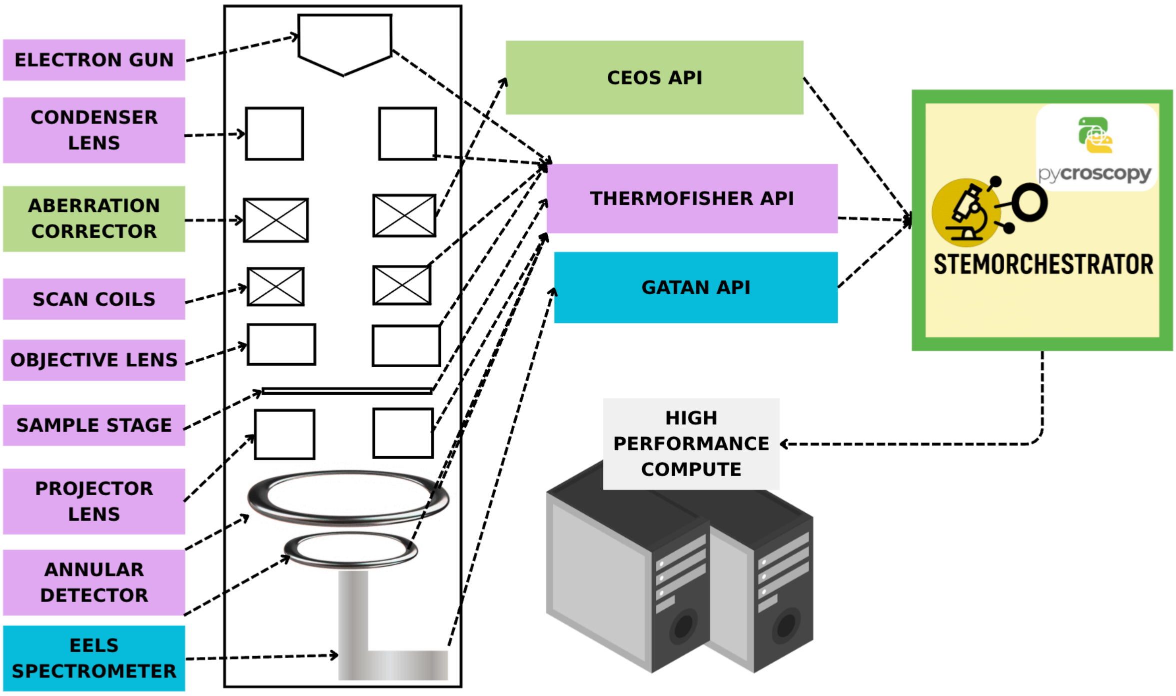


Figure 1: Illustration of the heterogeneous API control in a modern advanced STEM(Schematic). The schematic highlights how different hardware components of the microscope are controlled by disparate manufacturer APIs (ThermoFisher, CEOS, and Gatan).

Realizing the potential of machine learning and automated characterization in microscopy requires unified control over fragmented hardware APIs (**Figure 1**). To address this, we developed stemOrchestrator: a modular framework that integrates multiple STEM interfaces to facilitate sophisticated, automated workflows. We demonstrate the flexibility of the framework—though it is not limited to these examples—through three representative workflows: high-throughput particle characterization, hardware tuning using BO, and cross-correlation–based drift correction with informative logging of hardware status. Beyond these use cases, the framework is designed to support future integration of large language model (LLM) agents that can intervene, suggest, and execute complex automated workflows. The codebase is openly available for use and community contribution at: https://github.com/pycroscopy/pyAutoMic/tree/main/TEM/stemOrchestrator

We discuss how the package is designed in section **II** with emphasis on modularity, so it is **easily extensible** as an open-source effort. Section **III** discusses example automated workflow(non-exhaustive) it enables**. Section IV** then talks about the future of integration and development.

II. Software Implementation and Network Architecture

II. 1. Design Philosophy and Internal Logic

The internal logic of stemOrchestrator is designed to serve as a unified gateway to the microscope, bridging the critical gap between physical hardware and computational logic. While the architecture is necessitated by the sequential data flow of a typical STEM experiment, spanning Acquisition, Processing, Simulation, and Decision Making, the framework does not merely mirror this sequence; rather, it provides the essential "window" through which these distinct stages can interact seamlessly.

In a modern analysis workflow, a researcher must continually cycle through diverse tasks: configuring lenses and detectors, standardizing high-dimensional data streams, and comparing experimental outputs with simulations to make real-time decisions. stemOrchestrator acts as the programmable entrance to this environment. By abstracting the fragmented manufacturer APIs, it transforms the microscope into a cohesive entity accessible to software agents. This "window" allows complex, automated workflows, for example high-throughput particle characterization, active learning for hardware tuning, or adaptive drift correction, to drive the experiment directly, treating the instrument as a responsive component within a larger computational loop.

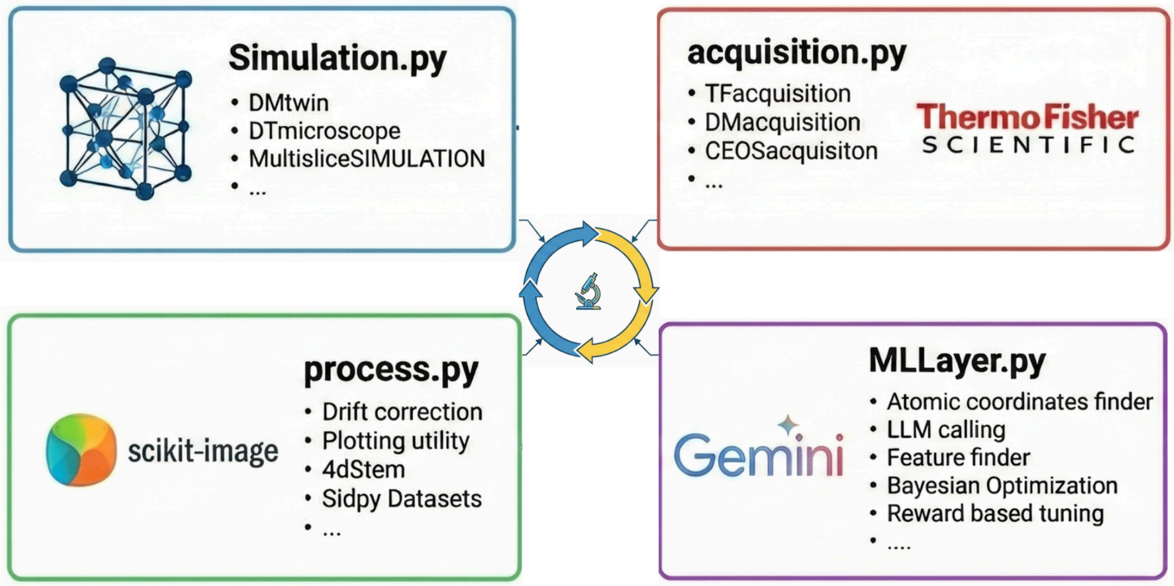


Figure 2. Visual block layout of stemOrchestrator showing the modularized design philosophy. The *acquisition.py* module is the central focus of this work (hardware control).

The codebase is structured into interoperable components to ensure extensibility and ease of debugging.

1. Hardware Abstraction and Protocol Management:

The foundational communication layer (hardware.py) abstracts the complexities of asynchronous network programming by utilizing the Twisted framework. We implemented the _CEOSProtocol class, which inherits from NetstringReceiver, to ensure data integrity by encapsulating JSON payloads within netstrings (length-prefixed byte strings). This design prevents TCP fragmentation issues common in high-frequency instrument polling. The CEOSacquisition class employs a deferred execution model where commands generate unique transaction IDs (self._next_id) and register Deferred objects in a pending dictionary, allowing the orchestrator to remain responsive to user interrupts while waiting for hardware acknowledgment. For high-speed operations where the overhead of the Twisted event loop is unnecessary, the CEOSacquisitionTCP class provides an alternative blocking socket connection with a 5-minute timeout, optimized for rapid, sequential command execution.

2. Acquisition Layer and Instrument State tracking:

The acquisition.py module serves as the primary interface for instrument state management and automated control routines. The TFacquisition class initializes interactions by verifying detector presence via the TemMicroscopeClient and handles the critical task of normalizing disparate hardware units, converting relative stage translations to nanometers and absolute positions to meters, while tilt angles are transformed between radians and millidegrees using helper methods like rad2miliDegree_stage. To ensure downstream models receive physically calibrated data, image acquisition routines return "Adorned" objects containing essential metadata such as pixel size and defocus rather than simple raw arrays. Beyond basic control, this layer implements intelligent workflows; for example, the jog_and_capture_with_metrics method executes step-wise stage movements while calculating image statistics in real-time, utilizing an early stopping mechanism if the metric declines consecutively to automate dataset creation.

3. Data Processing and Metadata Parsing:

The process.py module provides a unified toolkit for image normalization, metadata extraction, and registration. To ensure data provenance, the tiff_metadata function utilizes tifffile to parse standard tags and extract embedded instrument XML data (specifically under the "FEI_TITAN" tag) into nested Python dictionaries using xml.etree.ElementTree. This metadata supports accurate visualization, such as the HAADF_tiff_to_png function, which uses matplotlib_scalebar to draw physically accurate scale bars based on the extracted pixel size. The module also implements sub-pixel drift correction via compute_drift, which uses phase cross-correlation (skimage.registration.phase_cross_correlation), and verifies the results using plot_drift_comparison to generate composite views of the reference and shifted images.

4. Machine Learning Integration (MLlayer):

The Machine Learning Layer (MLlayer) is divided into specialized sub-modules for optimization and segmentation. The MLlayerBO.py module wraps packages such as BoTorch and GPyTorch to implement Gaussian Process (GP) regression, using classes like Tune1d, Tune2d, and Tune3d to manage the optimization loop. This loop iteratively maximizes the Expected Improvement (EI) acquisition function to select candidate hardware parameters for the next experimental step. Parallel to this, MLlayerSAM.py integrates the Segment Anything Model (SAM) for particle detection, preprocessing grayscale images into RGB format and extracting object boundaries using OpenCV's cv2.findContours. Crucially for automated targeting, the sample_particle_positions function applies a stratified sampling strategy to extract both centroids and a user-defined percentage of boundary points, facilitating precise beam positioning for spectroscopic analysis.

5. Simulation and Offline Development:

Finally, to ensure code reproducibility and facilitate offline development, the simulation.py module includes a DMtwin class. This digital twin mimics the API signatures of the real Gatan spectrometer, generating synthetic EELS spectra that include physically realistic features such as Zero Loss Peaks (ZLP), plasmon resonances, and background noise. This abstraction allows the entire orchestration pipeline to be tested and debugged without requiring immediate access to the physical microscope.

II. 2. Network Architecture and Communication Protocols

To bridge the gap between legacy instrument control PCs (often running Windows) and modern High-Performance Computing (HPC) environments where ML orchestration occurs, we implemented a multi-protocol communication layer. This design allows the main orchestration script to "import" the microscope as if it were a local library, completely hiding the complexity of the underlying network transport.

We utilized three distinct connection modalities, each optimized for specific hardware constraints:

For the CEOS aberration corrector, we implemented an asynchronous, event-driven messaging layer built on the Twisted framework. This architecture structures the communication payload using JSON-RPC over Twisted Netstrings, a method chosen specifically to define rigid message boundaries and prevent packet fragmentation. To ensure experimental reliability, the `_CEOSFactory` class within the hardware layer manages intrinsic reconnection logic, making long-duration data acquisition campaigns resilient to transient network drops or jitter.

In scenarios requiring high-frequency feedback, where the overhead of JSON parsing and event-loop scheduling is undesirable, we developed a lightweight direct control interface. This client communicates via raw TCP sockets, bypassing higher-level abstraction layers to interact directly with ports exposed by the server hardware. This "bare-metal" approach minimizes computational overhead, offering the lowest possible latency for critical control loops where millisecond-level timing is essential.

Finally, for instruments that are restricted to local Python APIs, specifically Thermo Fisher AutoScript and Gatan Digital Micrograph, we employed a remote object proxying strategy using wrappers such as Pyro5. Because these systems do not natively support network control, we wrap their local API objects to expose them over the network. This allows the central client (`acquisition.py`) to interact with a local proxy object that transparently handles the serialization and transmission of calls to the remote instrument PC, unifying disparate interfaces under a single control architecture.

II. 3. Security and Reproducibility

Given that instrument PCs often run older operating systems required by vendor software, the framework allows heavy computational logic to be offloaded to secure HPC environments. Users deploying this framework must ensure that the TCP/RPC ports used by *hardware.py* are firewalled and accessible only via a secure local VLAN or SSH tunnel.

To reproduce this work, researchers can use the modular design; specifically, the simulation.py module allows for the full execution of the *MLlayer.py* and *process.py* workflows using synthetic data, enabling development without immediate access to physical microscope hardware.

III. Example workflow enabled

In this section, we use stemOrchestrator to illustrate some of the most common workflows, demonstrate their implementation, and propose benchmark examples for evaluating performance and reproducibility.

IIIA. SAM-EDX-EELS-Diffraction

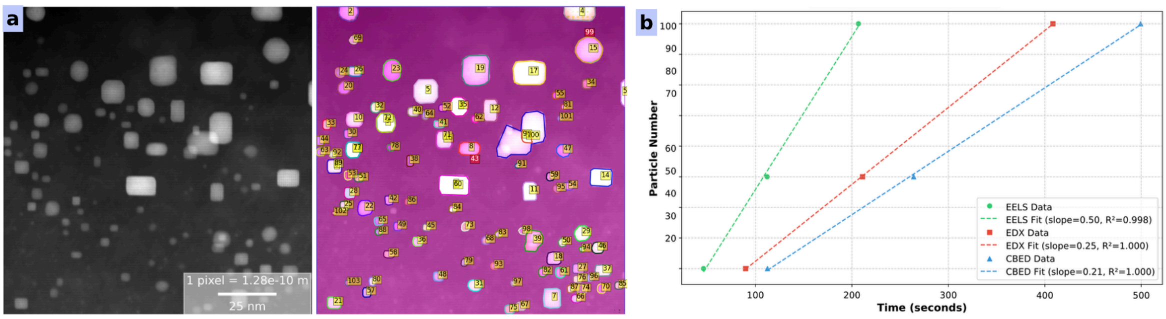


Figure 3. Automated particle analysis using HAADF-STEM imaging and multimodal spectroscopy. (a) Example segmentation maps showing over 100 particles, detected using the Segment Anything Model (SAM). (b) Acquisition times for Electron Energy Loss Spectroscopy (EELS), Energy-Dispersive X-ray Spectroscopy (EDX), and Diffraction across nine experimental runs at different particle counts. The near-linear scaling trend demonstrates the pipeline's efficiency and suitability for high-throughput characterization.

High-throughput particle characterization is a critical need in modern materials research, especially in studies involving heterogeneous nanoparticles and catalytic systems. Traditional Scanning Transmission Electron Microscopy (STEM)-based Electron Energy Loss Spectroscopy (EELS) workflows can take upwards of 10–12 hours to acquire spectrum images for just ~100 particles, often requiring custom scripts to selectively sample only at particle boundaries or centers to reduce overhead.

In this study, we present an accelerated, automated STEM workflow powered by the Segment Anything Model (SAM), enabling rapid, multimodal characterization of particles using Diffraction, Energy-Dispersive X-ray Spectroscopy (EDX), and EELS. As shown in **Figure 2 a**, our pipeline processes datasets with 20+, 50+, and 100+ particles across three magnifications. We demonstrate this capability on a TiO_2–Au sample, where Au nanoparticles are embedded in a TiO_2 matrix. The workflow includes pre-processing to enhance contrast, SAM-based segmentation for robust particle identification, and post-processing to extract morphological and statistical descriptors.

This approach bypasses the bottleneck of manual segmentation and spectrum image acquisition, allowing automated analysis with minimal human intervention. Benchmarking against manually curated datasets confirms that we maintain high accuracy while significantly improving throughput. Specifically, our system acquires EELS data for 100 particles in just ~3 minutes, EDX in ~7 minutes, and CBED Diffraction in ~10 minutes, scaling linearly with particle count, as illustrated in **Figure 2**.

The workflow is realized on a ThermoFisher Scientific (TFS) STEM instrument interfaced with the ISAACS supercomputer at the University of Tennessee, Knoxville (UTK). Hardware control and acquisitions (HAADF, EDX, Diffraction) were managed through ThermoFisher AutoScript, while EELS acquisition was automated via the Gatan DigitalMicrograph (DM) Server software using the Gatan Image Filter (GIF). Acquisition times per particle are as low as 0.1 seconds for CBED, 20 milliseconds for EELS, and 2 seconds for EDX. Importantly, this solution is portable and can be deployed on any AutoScript-enabled TFS microscope. Our work bridges deep learning and electron microscopy, providing a scalable, high-throughput solution for rapid characterization of nanomaterials.

III B. Drift correction

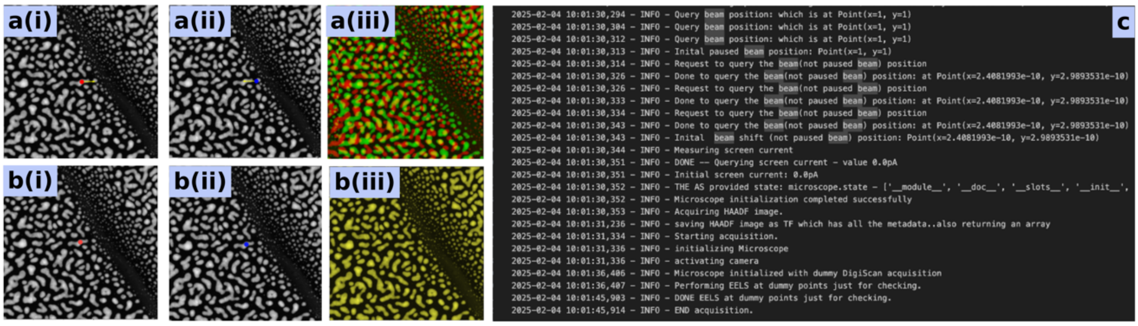


Figure 4. Panels (a) and (b) illustrate an example of the real-time drift correction workflow applied to High-Angle Annular Dark-Field (HAADF) images within the STEM Orchestrator. In (a), the stage was intentionally shifted to compute drift, resulting in a measured shift of 37.14 pixels in X and -0.43 pixels in Y. In (b), the verification step shows no drift (Image 1 = Image 2) with a measured shift of 0 pixels in both X and Y. Panel (c) displays an example of the live logging output, highlighting how various hardware components are actively queried during the drift correction process.

High-precision microscopy workflows often require dynamic interaction between ML agent and the hardware modules—querying an image, identifying features of interest, determining their coordinates, and then triggering spectroscopic acquisitions at those target locations. However, sample drift during this loop introduces spatial misalignments, which can significantly affect the accuracy of data collection—particularly in tasks requiring nanoscale or even sub-angstrom precision. To address this, we implemented an automated drift correction module within the STEM workflow. Our solution is based on cross-correlation between reference and current images, allowing us to estimate the drift vector and apply the appropriate correction—either by shifting the beam or repositioning the stage. This ensures that subsequent spectroscopic measurements are aligned with the originally selected target points, even if drift has occurred in the interim.

Drift can also be handled using predictive models that learn the underlying behavior of the stage or sample over time, which may be beneficial in high-drift or ultra-precise applications such as atomic-scale manipulation. While these approaches are part of our future roadmap, the current cross-correlation method offers a robust and generalizable solution for most use cases. As shown in Figure 3, this correction module has been integrated seamlessly into the real-time feedback loop of our automated STEM control stack.

III C. Bayesian Optimization and Active Learning for Experimental Automation using Rewards

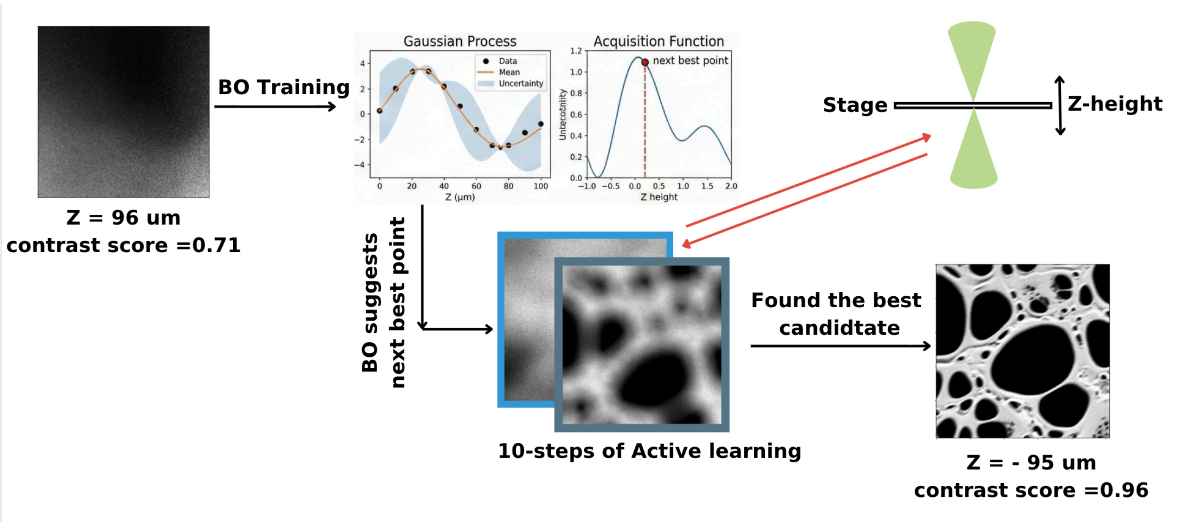


Figure 4. Using BO to tune the sample height of the STEM.

Experimental design in electron microscopy often involves selecting the next measurement point based on prior knowledge, intuition, or visual inspection. However, human decision-making is inherently sequential, potentially biased, and difficult to reproduce. Active learning provides a data-driven alternative by systematically suggesting where to measure next—balancing exploration and exploitation strategies. This can be formalized as a black-box optimization problem:

$$x^{\cdot} = arg\ \mathbf{max}_{x \in X}\ f(\mathrm{x}) \qquad (1)$$

where $f(x)$ is an expensive-to-evaluate function (e.g., quality of an image or spectrum), and x represents the experimental parameters (e.g., defocus, aberration coefficients, sample height). Since $f\ (x)$ is not known a priori and costly to sample, Bayesian Optimization (BO) offers an efficient framework to model it using a surrogate (typically a Gaussian Process, GP) and choose the next point via an acquisition function like Expected Improvement (EI), Upper Confidence Bound (UCB), or Entropy Search.

In microscopy, BO has proven useful in automating parameter tuning tasks. For example:

- BO can help find optimal corrector knob settings that minimize image distortion or maximize image sharpness based on sample height, stigmator coefficients or aberration coefficients values, as demonstrated in prior works like BEACON(Pattison et al., 2024).

- Spectroscopy acquisition (e.g., EELS): BO can guide the microscope to regions with higher likelihood of exhibiting desired features such as elemental peaks, edge plasmons and other desired features in the EELS spectrum(Pratiush, Roccapriore, et al., 2024).

As shown in **figure 4.** The Optimization problem is same as shown in eqn (1), where f(x) is the contrast measure at the different sample height. The sample height “x” can vary for -100 micrometer to +100 micrometer. The contrast measure is defined as

$$f(\mathbf{x}) = \frac{\sigma(\bar{I}(x))}{\mu(\bar{I}(x))} \tag{2}$$

equation 2 defines the objective function, which is being optimized shown in equation 1, $\bar{I}(x)$ is the normalized HAADF image acquired at sample height “x”, $\sigma(\bar{I}(x))$ is the calculated std deviation of the normalized image and $\mu(\bar{I}(x))$ is the mean of the normalized image.

Moreover, by leveraging Deep Kernel Learning (DKL), GPs can be made to operate on high-dimensional inputs (e.g., image patches or feature descriptors) enabling richer, context-aware acquisition policies. This integration of BO into real-time feedback loops unlocks new capabilities for adaptive experiments, enabling autonomous optimization of instrument parameters or data quality in a statistically principled and reproducible way.

III. D LLM assisted image understanding and orchestration.

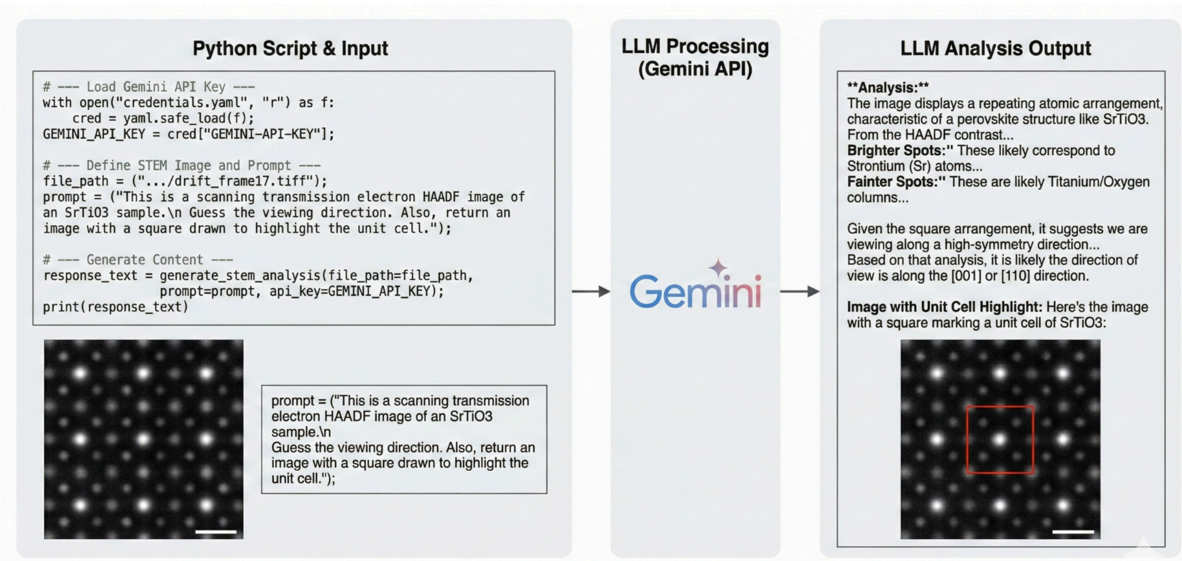


Figure 5. Using LLM in the experimental loop for image understanding.

Large Language Models (LLMs) offer a powerful new interface between microscopy images and actionable experimental insights. **Figure 5** shows an example validating zone axis alignment between literature and acquired image and deciding to tilt the stage to another desired zone axis. Beyond interpretation, LLMs act as orchestration assistants(to be added in future), capable of interfacing across the three layers (Hardware, ML and simulation and experiment logic) of the microscope control stack: Layer 1 (Hardware): LLMs can parse equipment manuals and provide quick guidance on commands, syntax, and expected behavior for hardware-level operations such as stage tilt or beam alignment. Layer 2 (Software & ML): They can recommend analysis workflows, pre-processing techniques, or model selection strategies tailored to the sample and imaging goals. Layer 3 (Experiment Logic): LLMs can suggest adaptive experiment plans, including candidate reward functions for optimization, e.g., maximizing image contrast, minimizing drift, or targeting edge-on interfaces in tomography.

IV. Future work

Building on the modular and extensible design of our current workflow, future efforts will focus on integrating this framework into the broader BlueSky ecosystem(Allan et al., 2019), enabling real-time feedback loops and multithreaded experimental orchestration in electron microscopy. We also envision human in the loop and incorporation of Large Language Model (LLM)-based agentic workflows, where intelligent agents can autonomously plan, execute, and adapt microscopy experiments based on literature insights, past data, and predefined scientific goals.

V. Conclusion

In summary, STEM-Orchestrator is a modular, open-source framework that unifies control of diverse STEM hardware, decoupling acquisition, simulation, processing, and ML layers into interoperable modules to enable rapid, high-throughput workflows. We demonstrated its effectiveness through particle mapping an AI-driven segmentation and multimodal spectroscopy pipeline that reduces acquisition times, and a real-time drift-correction module that maintains nanoscale alignment during extended experiments. By seamlessly integrating Bayesian optimization into live feedback loops. Its cohesive interface to beams, stages, detectors, and existing machine learning and electron microscopy codes not only accelerates materials discovery but also fosters collaboration and reproducibility across Machine learning, experiment and simulation communities. The framework's extensible design invites community contributions—whether adding new instrument backends, simulation engines, or advanced AI models and paves the way for next-generation, LLM-assisted agentic workflows. The source code is available for testing and extension at https://github.com/pycroscopy/pyAutoMic/tree/main/TEM/stemOrchestrator.

Acknowledgement

The work was partially supported (UP) AI Tennessee Initiative at University of Tennessee Knoxville (UTK). (U. P, A.H, G. D, S.V.K) acknowledges support from high performance computing facility, ISAAC and Microscopy Core Facility, of The University of Tennessee, Knoxville (UTK). This work was partially supported (AH, GD) by the U.S. DOE, Office of Science, Materials Sciences and Engineering Division and the Center for Nanophase Materials Sciences, which is a DOE Office of Science User Facility. The work was partially supported (SVK) by the Center for Advanced Materials and Manufacturing (CAMM), the NSF MRSEC center. The preprint of this is article is available at https://doi.org/10.31224/4645

References

ABEBE, W., STRUBE, J., GUO, L., TALLENT, N. R., BEL, O., SPURGEON, S., DOTY, C. & JANNESARI, A. (2025). SAM-I-Am: Semantic boosting for zero-shot atomic-scale electron micrograph segmentation. *Computational materials science* **246**, 113400.

ALLAN, D., CASWELL, T., CAMPBELL, S. & RAKITIN, M. (2019). Bluesky's Ahead: A Multi-Facility Collaboration for an *a la Carte* Software Project for Data Acquisition and Management. *Synchrotron Radiation News* **32**, 19–22.

ARCHIT, A., FRECKMANN, L., NAIR, S., KHALID, N., HILT, P., RAJASHEKAR, V., FREITAG, M., TEUBER, C., BUCKLEY, G., VON HAAREN, S., GUPTA, S., DENGEL, A., AHMED, S. & PAPE, C. (2025). Segment Anything for Microscopy. *Nature methods* **22**, 579–591.

BARAKATI, K., LIU, Y., PRATIUSH, U., SLAUTIN, B. N. & KALININ, S. V (2025). Rewards-based image analysis in microscopy. https://arxiv.org/abs/2502.18522.

BOTIFOLL, M., PINTO-HUGUET, I. & ARBIOL, J. (2022). Machine learning in electron microscopy for advanced nanocharacterization: current developments, available tools and future outlook. *Nanoscale Horizons* **7**, 1427–1477.

CHERUKARA, M. J. (2024). High Performance Computing and Artificial Intelligence Enabled Materials Characterization and Experimental Automation. *Microscopy and Microanalysis* **30**, ozae044.1006. https://doi.org/10.1093/mam/ozae044.1006.

CHRISTOPHER, J., TALEB, M., MAITY, A., HENTSCHEL, M., GIESSEN, H. & TALEBI, N. (2020). Electron-driven photon sources for correlative electron-photon spectroscopy with electron microscopes. *Nanophotonics (Berlin, Germany)* **9**, 4381–4406.

COLLIEX, C., KOCIAK, M. & STÉPHAN, O. (2016). Electron Energy Loss Spectroscopy imaging of surface plasmons at the nanometer scale. *Ultramicroscopy* **162**, A1–A24.

COLÓN, Y. J. & SNURR, R. Q. (2014). High-throughput computational screening of metal-organic frameworks. *Chemical Society reviews* **43**, 5735–5749.

CREWE, A. V. (1974). Scanning transmission electron microscopy*. *Journal of Microscopy* **100**, 247–259.

DAGLAR, H. & KESKIN, S. (2020). Recent advances, opportunities, and challenges in high-throughput computational screening of MOFs for gas separations. *Coordination chemistry reviews* **422**, 213470.

EGERTON, R. F. (2011). *Electron Energy-Loss Spectroscopy in the Electron Microscope*. Third edition. New York, NY: Springer.

GREENAWAY, R. L., SANTOLINI, V., BENNISON, M. J., ALSTON, B. M., PUGH, C. J., LITTLE, M. A., MIKLITZ, M., EDEN-RUMP, E. G. B., CLOWES, R., SHAKIL, A., CUTHBERTSON, H. J., ARMSTRONG, H., BRIGGS, M. E., JELFS, K. E. & COOPER, A. I. (2018). High-throughput discovery of organic cages and catenanes using computational screening fused with robotic synthesis. *Nature communications* **9**, 2811–2849.

HETHERINGTON, C. (2004). Aberration correction for TEM. *Materials Today* **7**, 50–55.

KALININ, S. V., MUKHERJEE, D., ROCCAPRIORE, K., BLAISZIK, B. J., GHOSH, A., ZIATDINOV, M. A., AL-NAJJAR, A., DOTY, C., AKERS, S., RAO, N. S., AGAR, J. C. & SPURGEON, S. R. (2023). Machine learning for automated experimentation in scanning transmission electron microscopy. *npj Computational Materials* **9**, 227.

KALININ, S. V, OPHUS, C., VOYLES, P. M., ERNI, R., KEPAPTSOGLOU, D., GRILLO, V., LUPINI, A. R., OXLEY, M. P., SCHWENKER, E., CHAN, M. K. Y., ETHERIDGE, J., LI, X., HAN, G. G. D., ZIATDINOV, M., SHIBATA, N. & PENNYCOOK, S. J. (2022). Machine learning in scanning transmission electron microscopy. *Nature Reviews Methods Primers* **2**.

MADSEN, J. & SUSI, T. (2023). *ab* TEM: A Fast and Flexible Python-based Multislice Simulation Package for Transmission Electron Microscopy. *Microscopy and Microanalysis* **29**, 680–680.

MANLY, C. J., LOUISE-MAY, S. & HAMMER, J. D. (2001). The impact of informatics and computational chemistry on synthesis and screening. *Drug Discovery Today* **6**, 1101–1110.

MEIROVITCH, Y., PARK, C. F., MI, L., POTOCEK, P., SAWMYA, S., LI, Y., WU, Y., SCHALEK, R., PFISTER, H., SCHOENMAKERS, R., PEEMEN, M., LICHTMAN, J. W., SAMUEL, A. D. T. & SHAVIT, N. (2023). SmartEM: machine-learning guided electron microscopy. *bioRxiv*. https://www.biorxiv.org/content/early/2023/10/08/2023.10.05.561103.

PATTISON, A. J., RIBET, S. M., NOACK, M. M., VARNAVIDES, G., PARK, K., KIRKLAND, E., PARK, J., OPHUS, C. & ERCIUS, P. (2024). BEACON – Automated Aberration Correction for Scanning Transmission Electron Microscopy using Bayesian Optimization. https://arxiv.org/abs/2410.14873.

PENNYCOOK, S. J. , N. P. D. (2011). *Scanning Transmission Electron Microscopy*. Pennycook, S. J. & Nellist, P. D. (Eds.). New York, NY: Springer New York.

PRATIUSH, U., DUSCHER, G. & KALININ, S. (2024). Human-in-the-loop interface for Automated experiments in Electron Microscopy, Automated characterization. In *AI for Accelerated Materials Design - NeurIPS 2024*. https://openreview.net/forum?id=AB4EuiXkAr.

PRATIUSH, U., HOUSTON, A., KALININ, S. V. & DUSCHER, G. (2024). Realizing smart scanning transmission electron microscopy using high performance computing. *Review of Scientific Instruments* **95**.

PRATIUSH, U., ROCCAPRIORE, K. M., LIU, Y., DUSCHER, G., ZIATDINOV, M. & KALININ, S. V. (2024). Building Workflows for Interactive Human in the Loop Automated Experiment (hAE) in STEM-EELS.

PYCROSCOPY (2025). pyAutoMic: DigitalMicrograph TEM Scripts.

SPURGEON, S. R., OPHUS, C., JONES, L., PETFORD-LONG, A., KALININ, S. V., OLSZTA, M. J., DUNIN-BORKOWSKI, R. E., SALMON, N., HATTAR, K., YANG, W.-C. D., SHARMA, R., DU, Y., CHIARAMONTI, A., ZHENG, H., BUCK, E. C., KOVARIK, L., PENN, R. L., LI, D., ZHANG, X., MURAYAMA, M. & TAHERI, M. L. (2021). Towards data-driven next-generation transmission electron microscopy. *Nature Materials* **20**, 274–279.

THERMO FISHER SCIENTIFIC (2025). AutoScript TEM Software.

UTKARSHP1161 (2024). utkarshp1161/Active-learning-in-microscopy: v1.0.0. https://doi.org/10.5281/zenodo.14562670.

WELBORN, S. S., HARRIS, C., RIBET, S. M., VARNAVIDES, G., OPHUS, C., ENDERS, B. & ERCIUS, P. (2024). Streaming Large-Scale Microscopy Data to a Supercomputing Facility. *Microscopy and Microanalysis* **31**, ozae109. https://doi.org/10.1093/mam/ozae109.

WILLIAMS, D. B. & CARTER, C. B. (1996). The Transmission Electron Microscope. In *Transmission Electron Microscopy*, pp. 3–17. Boston, MA: Springer US.

ZIATDINOV, M., GHOSH, A. & KALININ, S. V. (2021). Physics makes the difference: Bayesian optimization and active learning via augmented Gaussian process.

ZIATDINOV, M., GHOSH, A., WONG, C. Y. & KALININ, S. V. (2022). AtomAI framework for deep learning analysis of image and spectroscopy data in electron and scanning probe microscopy. *Nature Machine Intelligence* **4**, 1101–1112.

ZIATDINOV, M., LIU, Y., MOROZOVSKA, A. N., ELISEEV, E. A., ZHANG, X., TAKEUCHI, I. & KALININ, S. V. (2021). Hypothesis Learning in Automated Experiment: Application to Combinatorial Materials Libraries.