Real-Time Calcium Imaging Analysis & Stimulation Platform

Raymond Chen[1], Anne Draelos[1,2], John Pearson[1,2]

1. Center for Cognitive Neuroscience, Duke University
2. Department of Biostatistics and Bioinformatics, Duke University

Background

Calcium imaging experiments can generate terabytes of data, coming at the cost of increased processing time. However, improvements in preprocessing algorithms and tools for characterizing neuron activity have allowed the design of a real-time analysis platform to support adaptive experiments.

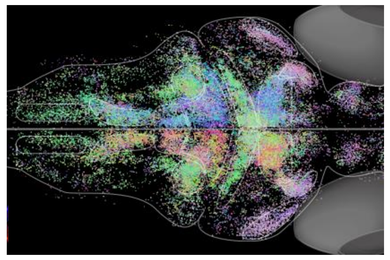


Fig. 1 | Zebrafish brain, highlighted motion-sensitive neurons

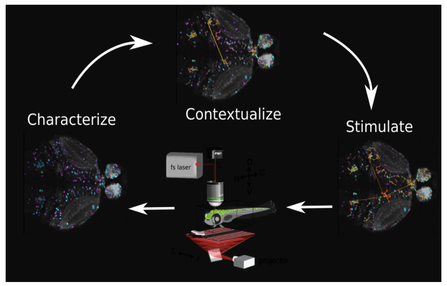


Fig. 2 | Closed-loop pipeline concept. Imaging data is acquired, processed, and visualized in real-time, allowing for targeted stimulation to test causal hypotheses.

Methods

We implemented our platform using modular components in Python with standard libraries and the CaImAn defined Anaconda environment. Modules execute and monitor activity involving data acquisition, processing, analysis, and visualization. Windows Subsystem for Linux was used to solve Windows compatibility issues with Apache Arrow.

- CaImAn → Open-source calcium imaging processing library[1]
- Multiprocessing → Library for concurrent Python processes
- Apache Arrow → Plasma in-memory object store

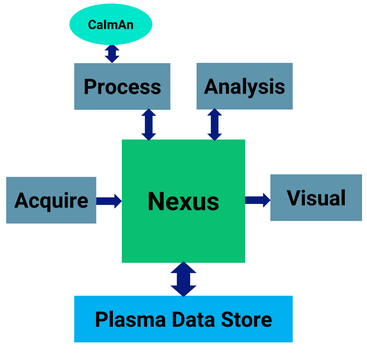


Fig. 3 | System Architecture

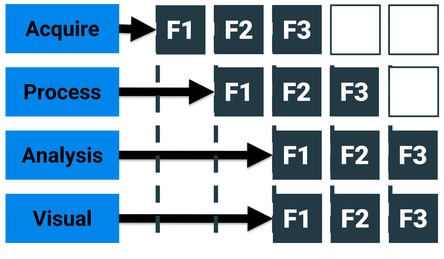


Fig. 4 | Modules run in parallel

Results

Our software platform is available for installation on Linux, Mac OS X, and Windows systems, and is designed to be setup and run with ease.

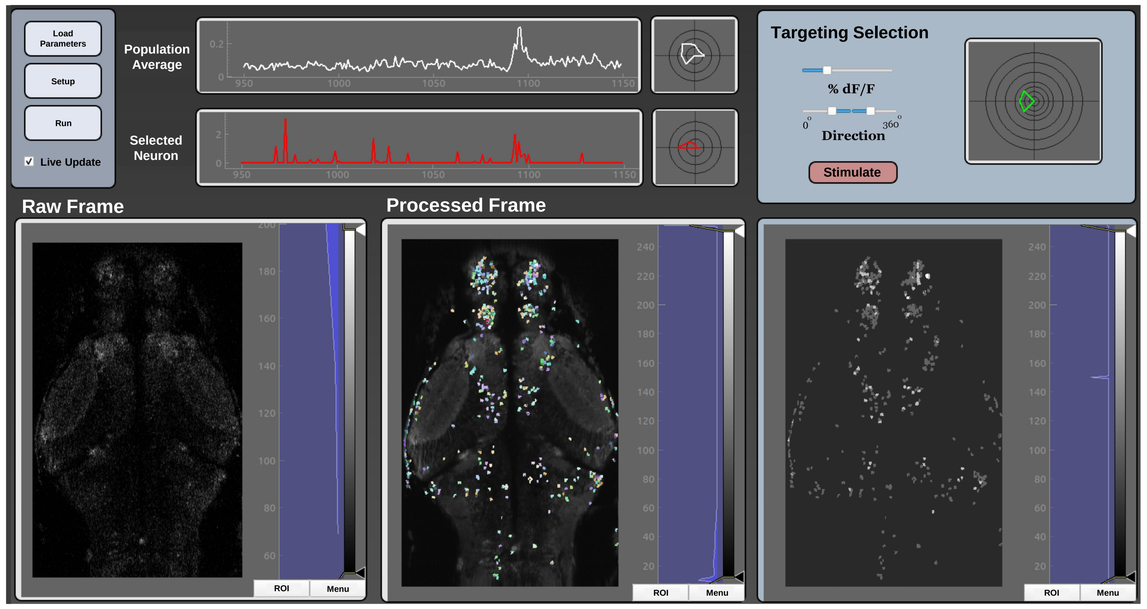


Fig. 5 | The user interface displays both raw and processed images in real-time, while also showing the full history of average and select neuronal activity over an entire experiment. In addition, there are panels to select neurons for stimulation.

Fig. 6 | Analysis of Plasma store performance during platform execution. When objects are stored and retrieved in a repeated fashion, performance appears to be constant. However, store timings do spike at reproducible, exponentially increasing counts.

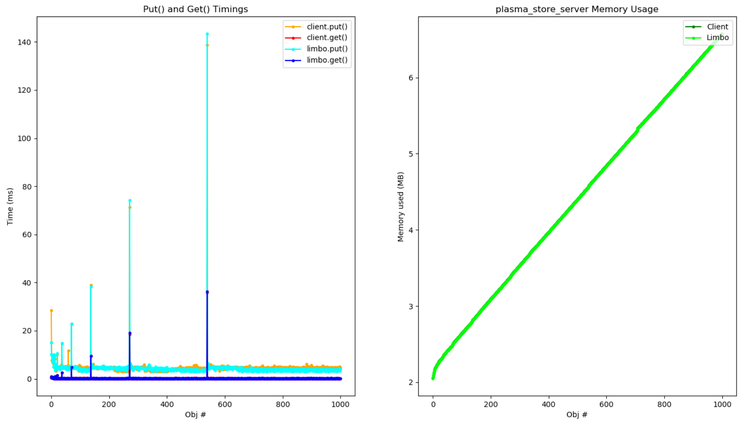
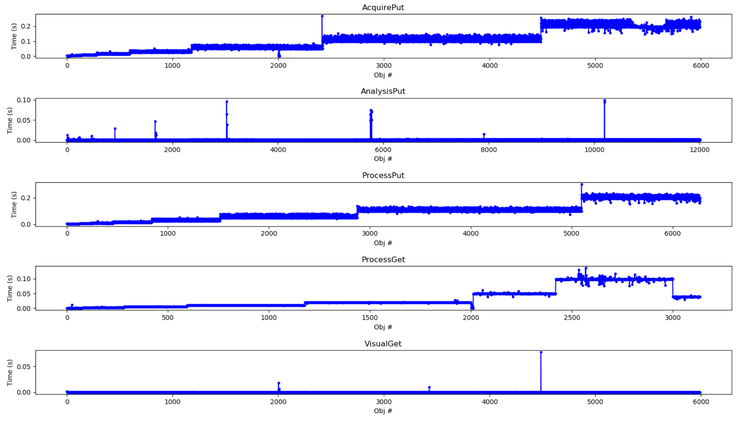


Fig. 7 | Plasma store access performance for individual modules. Interestingly, some modules have flat performance, while others have scaling times in a step fashion. We believe this is due to storing and retrieval order and frequency that is specific to certain modules.



Conclusion

This software platform is still in development. We hope to improve plasma store access efficiency, implement Just-in-time compilation for numpy code, improve the user interface, and begin testing on experiments soon. Nonetheless, our tool has demonstrated effective real-time processing and analysis of calcium imaging data that is executable on even some low-end computers.

References & Acknowledgements

1. Giovannucci, A., Friedrich, J., Gunn, P., Kalfon, J., Brown, B. L., Koay, S. A., . . . Pnevmatikakis, E. A. (2019). CaImAn an open source tool for scalable calcium imaging data analysis. ELife, 8. doi:10.7554/elife.38173

Thanks to Dr. Anne Draelos and Dr. John Pearson for their mentorship. Thanks to Dr. Andrew Huang and the Huang Fellows Program for their funding and support this summer. This research was supported by a DIBS Incubator Award.