bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

CalTrig: A GUI-Based Machine Learning Approach for Decoding Neuronal Calcium Transients in Freely Moving Rodents

Michal A. Lange[1], Yingying Chen[1], Haoying Fu[1], Amith Korada[1],

Changyong Guo[1*], Yao-Ying Ma[1,2*]

[1], Department of Pharmacology and Toxicology, Indiana University School of Medicine, Indianapolis, IN 46202, USA.

[2], Stark Neurosciences Research Institute, Indiana University School of Medicine, Indianapolis, IN 46202, USA.

***Corresponding authors**: Dr. Yao-Ying Ma & Dr. Changyong Guo

Department of Pharmacology and Toxicology, Indiana University School of Medicine

635 Barnhill Drive, Indianapolis, IN 46202

Tel: 317-274-1536

Fax: 317-274-7714

Email: ym9@iu.edu

KEY WORDS: *in vivo* calcium imaging, miniScope, calcium transients, machine learning, GRU, data visualization

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

ABSTRACT

Advances in *in vivo* Ca^{2+} imaging using miniatured microscopes have enabled researchers to study single-neuron activity in freely moving animals. Tools such as MiniAN and CaImAn have been developed to convert Ca^{2+} visual signals to numerical information, collectively referred to as CaIV2N. However, substantial challenges remain in analyzing the large datasets generated by CaIV2N, particularly in integrating data streams, evaluating CaIV2N output quality, and reliably and efficiently identifying $^{Ca2+}$ transients. In this study, we introduce CalTrig, an open-source graphical user interface (GUI) tool designed to address these challenges at the post-CaIV2N stage of data processing. CalTrig integrates multiple data streams, including Ca^{2+} imaging, neuronal footprints, Ca^{2+} traces, and behavioral tracking, and offers capabilities for evaluating the quality of CaIV2N outputs. It enables synchronized visualization and efficient Ca^{2+} transient identification. We evaluated four machine learning models (i.e., GRU, LSTM, Transformer, and Local Transformer) for Ca^{2+} transient detection. Our results indicate that the GRU model offers the highest predictability and computational efficiency, achieving stable performance across training sessions, different animals and even among different brain regions. The integration of manual, parameter-based, and machine learning-based detection methods in CalTrig provides flexibility and accuracy for various research applications. The user-friendly interface and low computing demands of CalTrig make it accessible to neuroscientists without programming expertise. We further conclude that CalTrig enables deeper exploration of brain function, supports hypothesis generation about neuronal mechanisms, and opens new avenues for understanding neurological disorders and developing treatments.

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

1. INTRODUCTION:

Neurons within the same brain region are diverse in type, connectivity, and activity, responding to stimuli with high temporal precision. One of the key advances in modern neuroscience is the shift from studying brain regions as functional units to focusing on individual neurons. Monitoring single-neuron activity in freely moving animals allows for deeper insights into brain function and dysfunction. Recent advances in *in* vivo imaging and fluorescent Ca^{2+} indicators, coupled with miniature microscopes (miniScopes), have revolutionized the study of neural dynamics in freely moving animals. Tools have been developed to convert Ca^{2+} visual signals (i.e., Ca^{2+} imaging video files) to numerical information (e.g., changes of fluorescent signal intensity indicating the Ca^{2+} influx for each neuron), denoted **CalV2N**. Early tools such as Suite2P[1], SIMA [2], STNeuronNet [3], CalmAn [4], were created to process two-photon imaging data. Due to the low resolution and poor signal to noise ratio (SNR) in one-photon imaging data, more specific CalV2N tools were created including MIN1PIPE[5] and MiniAn[6]. The algorithms used in CalV2N tools started with principal-component analysis/independent component analysis (PCA/ICA)[7], and then upgraded to constrained non-negative matrix factorization (CNMF), and its various derivations such as CNMF-E[6]. Due to better reliability in demixing the activities of overlapping cells, the computational efficiency and accessibility of parameter adjustment, CNMF-E based MiniAn has become a reliable choice and selected as the CalV2N tool to extract Ca^{2+} traces in this study.

Despite the availability of CalV2N tools, substantial challenges remain in drawing meaningful conclusions from Ca^{2+} imaging data. **First**, the lack of synchronized

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

visualization of multiple data streams: Usually at least two original data streams are collected, including behavioral video tracking and the Ca^{2+} image. After data processing by the CalV2N tool, three data streams are generated, including extracted Ca^{2+} traces, footprint of identified cells, and the processed video. The current available CaV2N tools are usually Python-based. Visualization of different lines of data or videos are primarily contained in discrete sections (e.g., background removal, seed selection, and other sections in MiniAn serving as an example CalV2N tool). Although this provides insight into the effects of single parameters, an integrative visual platform is missing that would provide the user with a review of the multiple lines of data. **Second**, the global parameters applied across the data often result in inconsistencies, as signal quality varies between neurons. The parameters that are available to the user are applied across the entirety of the input data, which can contribute to mixed results to the quality of extracted Ca^{2+} traces. MiniAn for instance, visualizes 5-10 cells at random to ascertain the impact of a parameter. In our research group, we observed inconsistencies arising from differing SNRs across different cells and varying intensities of resultant Ca^{2+} traces, which has created a demand for a post-CalV2N tool. **Third**, the verification of detected cells and extracted Ca^{2+} traces remains uncertain. **Forth**, there is no well-established tool available to time efficiently and reliably identify Ca^{2+} transients. **Fifth**, we would also like to highlight that the challenges mentioned are significantly amplified by the complexity and sheer volume of data processed through CalV2N. This includes: (1) high temporal resolution, with sampling rates of 10-60 Hz; (2) high spatial resolution, with pixel sizes ranging from 0.8-1.0 μm and a field of view (FOV) up to 1.0 mm × 0.8 mm; (3) high cell throughput, with 50-200 neurons typically

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

detected per animal, resulting in massive datasets, especially when scaled across multiple animals (e.g., 10-20 per group) and various experimental conditions; and (4) integration with behavioral data, adding further complexity to the analysis.

To address these challenges at the post-CalV2N stage, we developed the Ca^{2+} transient identifier GUI (CalTrig), a graphical user interface (GUI) using the Python package PyQT5. CalTrig integrates all outputs from MiniAn (including Ca^{2+} imaging original processed videos, Ca^{2+} traces, cell footprints, and behavioral videos) to facilitate synchronized visualization, evaluate the performance of CalV2N, and identify Ca^{2+} transients. The objective of creating CalTrig is listed below:

- **Efficient performance**: Low computing demands ensure smooth operation on standard computers.
- **User-friendly interface**: Neuroscientists without programming skills can explore and analyze data at neuronal levels with high temporal resolution.
- **Integrative visual exploration**: CalTrig enables evaluation of CalV2N performance and reliable detection of Ca^{2+} transients by integrating multiple data streams and videos.
- **Multiple options for Ca^{2+} transient detection**: CalTrig supports manual, parameter-based, and machine learning-based detection methods.
- **Integration of detection strategies**: Detection strategies can be combined. For example, parameter-based autodetection or machine learning outputs can be manually corrected, and parameter-based autodetection can serve as a preliminary

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

step before manual detection, helping set up datasets for training machine learning models.

- **Informative output**: Detailed Ca^{2+} transient data for each neuron and an overview of a pool of neurons from one mouse brain can be exported as data tables or figures for publication.

In summary, CalTrig is developed to bridge the gap between CaIV2N tools and the final stages of data analysis, simplifying the workflow and enabling researchers to extract meaningful biological insights from raw data. Although the potential readership is broad, this research article primarily aims to assist neurobiologists in processing *in vivo* Ca^{2+} imaging data collected using a single-photon miniScope. We would like to give a brief introduction to the article's organization, which differs from a typical neurobiological research paper. In the **Introduction**, we elaborate on the significance of *in vivo* Ca^{2+} imaging in freely-moving animals and discuss the challenges encountered after extracting Ca^{2+} traces using tools like CaIV2N and MiniAn. We then concluded the goals for developing CalTrig. The **Methods** section not only reiterates previously established procedures for collecting Ca^{2+} imaging data from freely-moving mice and extracting the Ca^{2+} traces, as we reported earlier, but also provides detailed information about the newly developed tool, CalTrig. The latter includes data loading from CaIV2N, data visualization, cell verification, CaIV2N evaluation, Ca^{2+} transient identification, and exporting figures or data for statistical analysis, ready for publication. We present three strategies for Ca^{2+} transient identification (i.e., parameter-based, manual, and machine learning-based detection), including their procedures and applications. In the **Results** section, we compare the performance of multiple machine learning models, evaluate whether the established machine learning model can be used to identify transients across datasets collected at different time points, from different animals, and in various brain regions. Finally, the

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Discussion highlights the advantages of using machine learning models, the integrative visual exploration interface, and the unique features of CalTrig.

2. METHODS:

2.1. *In vivo* Ca^{2+} Imaging data collection

2.1.1. Experimental animals

All in vivo procedures on laboratory animals were performed in accordance with the United States Public Health Service Guide for Care and Use of Laboratory Animals and were approved by the Institutional Animal Care and Use Committee at Indiana University School of Medicine. Ten male C57BL/6J, bred in-house using breeders originally derived from the Jackson Laboratory, were used in this study. The mice were group housed except for those undergoing GRIN lens implantation, which were singly housed to prevent cage mates from damaging the implanted lens. All mice had free access to chow and water in home cage, maintained on a 12-hour light/dark cycle (light on at 7:00 AM and off at 7:00 PM).

2.1.2. Surgical procedures:

Microinjection of AAV

Mice were anesthetized with 2.5% isoflurane for induction and maintained with ~1.2%. A 28-gauge injection needle was used to unilaterally inject the tAAV1-Syn-jGCaMP8f-WPRE or AAV1-CaMKIIa-jGCaMP8f-WPRE solution (0.5 µl/site, 0.1 µl/min) *via* a Hamilton syringe into the M2 (coordinates in mm: AP, +1.80; ML, ±0.60; DV, -1.30 for M2), using a Pump 11 Elite Syringe Pumps (Harvard Apparatus). Injection needles were left in place for 5 min following injection.

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Lens implantation

A couple of minutes after withdrawing the AAV injection needle, a unilateral GRIN lens (Inscopix Inc, #1050-004595, Diameter: 1.0 mm; length: ~4.0 mm; Working Distance: 200 μm) was lowered through the cranial window to 200 μm above the center of the virus injection site. The open space between the lens' side and the skull opening was sealed with surgical silicone (Kwik-Sil) and secured by dental cement (C&B Metabond). The exposed part of the lens above the skull was further coated with black cement (Lang Dental Mfg. Co.' Inc.).

Base-plating

After the catheter implantation, the mouse was maintained anesthetized with isoflurane. The cement on top of the GRIN lens was carefully removed using drill bits until the lens was exposed. The top of the lens was then cleaned using lens paper and a cleaning solution. A metal baseplate was mounted onto the skull over the lens using Loctite super glue gel, guided by a MiniScope for optimal field of view. Once the baseplate was securely mounted, the MiniScope was removed. A protective cap was attached to the baseplate, and the mouse was returned to its home cage.

Verification of AAV expression and the lens location

After the completion of the in vivo Ca^{2+} imaging, M2 or mPFC-containing coronal slices were prepared as described before[8], then fixed in 4% Paraformaldehyde (PFA) for no less than a couple of hours. After a brief rinse with PBS, slices were mounted with Prolong^{TM} Gold antifade mounting reagent with DAPI (Invitrogen, Cat# P36931). Confocal

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

imaging was performed using a Zeiss LSM 800 confocal microscope. The criteria for animal inclusion in this study were (a) highly enriched AAV expression in mostly pyramidal neurons, within-M2 or mPFC viral injection site, and (b) the footprint of the GRIN lens tip at the top of the targeting brain area.

2.1.3. In vivo Ca^{2+} imaging recording

Mice were habituated to the *in vivo* Ca^{2+} recording procedure by mounting the miniScope V4 (OpenEphys) to the pre-anchored baseplate and recording for 5 min per day at home cage for 3 days before starting the 1-hr daily recordings in an operant chamber (Med Associates). Data Acquisition (DAQ) box, supported by an open source, C^{++} and Open Computer Vision (OpenCV) libraries-based software, were used to collect both Ca^{2+} and behavioral video streams simultaneously controlled by the operant chamber software, MED-PC (Med Associates) *via* a TTL adaptor. The sampling frequency was 30 Hz.

2.2. Extraction of Ca^{2+} transient traces from the raw videos

Among multiple types of computational tools established previously to extract Ca^{2+} transients from raw videos, a Python-based analysis pipeline, Minian[9], was used in our data analyses due to its low memory demand and user friendly parameter options. In brief, there were five steps in the pipeline. First, multiple raw videos were batch loaded and subjected to a *PREPROCESSING* stage, where sensor noise and background fluorescence from scattered light were removed. Second, rigid brain motions were corrected by *MOTION CORRECTION*. Third, the initial spatial and temporal matrices for later steps were generated by a seed-based approach, called *SEEDS INITIALIZATION*.

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Fourth, the spatial footprints of cells were further refined. Fifth, the temporal signals of cells were also refined. The last two steps, the *SPATIAL UPDATE*, and the *TEMPORAL UPDATE*, as the core computational components based on CNMF algorithm, were repeated at least one more time.

2.3. CalTrig (Ca^{2+} Transient identification GUI)

CalTrig is python-based open source code with GUI interface. It is available in our lab Github station (webpage to be added). The repository contains documentation, demos, and a message/discussion board. The code, which is compatible with Python 3.10, uses several open-source libraries including Xarray, Numpy, Pandas, PyQt5 and Pyqtgraph.

Computer system specifications for CalTrig development and testing: The development and testing of CalTrig were conducted on a system with the following specifications: CPU, Ryzen 9 7900X; RAM, 64GB; GPU, Nvidia RTX 4080; Operating System, Windows 11.

2.3.1. Data loading

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

The front page of CalTrig serves as a hub for loading data (**Fig. S1A**). Data can be loaded by directly specifying the file path, or indirectly by loading the pre-generated INI (Initialization) files or JSON (JavaScript Object Notation) files.

Table 1. CNMF variable

CNMF variable	Definition
C	Temporal matrix of Ca^{2+} dynamics
S	Temporal matrix of Ca^{2+} transient rising slope
A	Spatial matrix of cell footprints
B	Spatial matrix of background footprints
f	Temporal vector of background dynamic.
YrA	Temporal matrix of residual cell traces.
Y/....	A collection of 3D matrices containing original and processed videos.
ΔF/F	Temporal matrix of ΔF/F values for cells. This is the only variable that is not considered mandatory, as the MiniAn pipeline does not generate it. When absent, it will be automatically generated in the CalTrig using a CAIMAN implementation.

The INI file uses a flat format with itemized information to detail a single data set. It incorporates both basic experimental design details (e.g., Animal ID, day, session stage) and multiple temporally synchronized data lines, including (1) the original and CNMF process Ca^{2+} image video, (2) extracted Ca^{2+} trace data (i.e., the CNMF-E output), (3) behavioral data (i.e., operant behavioral such as active lever press, ALP; inactive level press, ILP; and Reinforcement, RNF), and (4) behavioral video tracking. The variables included in each CNMF dataset are listed in **Table 1** and included in Parameter list in CalTrig interface (**Fig. S2**). The INI file can be created by following the format provided in demo INI files, ensuring that both the experimental design and data streams are properly synchronized for exploration and analysis.

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

The JSON file, in contrast, uses a hierarchical format, representing the experimental design. Using data from one of our addiction projects as an example, the JSON file can collect information from multiple INI files, outlining details such as treatments for each animal, Ca^{2+} imaging days, and session stages. The JSON file can be generated by saving data after loading multiple INI files, either for longitudinal recordings in a single animal or for a group of animals assigned to the same experimental group.

2.3.2. Data visualization

As shown in **Fig. S1B**, each set of data can be visualized in the window by their cell footprint and are arranged in a grid related to the experiment design details, such as animal ID, day, session stage. Select the data set to be explored by clicking the corresponding window of the footprint, then click the “cell exploration” button to open a separate window that is subdivided into five main components (**Fig. 2**), including

(1) Ca^{2+} image video: toggle between original vs. CNMF processed videos, can be zoomed in to see more details about the cellular signal.

(2) behavioral video tracking, showing behaviors of the freely-moving animal during Ca^{2+} imaging recording,

(3) cell list, including individual cell# identified by CNMF-E,

(4) Ca^{2+} trace window,

(5) Trace toolbox

The following tasks can be done by the crosstalk between windows:

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

- Footprint of cells of interest (COI): The footprint of the detected neurons can be visualized by clicking the projection area in the Ca^{2+} imaging video, or by selecting the cell number from the cell list. This action superimposes its corresponding footprint onto the Ca^{2+} imaging video. The footprint can be displayed as a solid patch, a contour or by dimming the non-cell intensity.
- Ca^{2+} traces of cells of interest: The cells of interest can be selected by clicking on the footprint area in Ca^{2+} imaging video, or selection of the cell# in the cell list.
- Variable readouts of Ca^{2+} traces: Ca^{2+} traces can be visualized with different temporal readouts as listed in Trace toolbox, including the variables directly imported from CNMF-E (i.e., C signal and S signal), the variables calculated by CalTrig (i.e., ΔF/F, Raw signal, SavGol Filter of ΔF/F, Noise, SNR).
- Time stamp of behavioral events or external stimuli on Ca^{2+} traces: Ca^{2+} traces can be segmented by the behavioral readout listed in the Trace toolbox, such as ALP, ILP, RNF,
- Standardized window size of the Ca^{2+} trace window: The window size for X-axis in the unit of frame, and the scale range for Y-axis, can be defined in this box. When clicking “Reset view” button, the magnification of the signal will be adjusted according to the Y axis range, and the window will be set up with the pre-defined size by anchoring the first visible frame in the Ca^{2+} trace window. This is considerably helpful when manually identifying Ca^{2+} transients.
- Cell verification: Cells identified by CNMF-E can be individually reviewed and categorized into “Approved cells”, “Rejected Cells” and Missed Cells” as detailed below by crosstalk between multiple windows.

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

- Quality evaluation of CNFM analysis:
- Ca^{2+} transient identification: this can be done by parameter-based auto detection, manual detection, machine learning-based detection, or combination of multiple strategies we have developed in CalTrig.

2.3.3. Cell verification and CalV2N evaluation

Cell verification

- Verified cell: CalTrig allows user to inspect individual cells detected by CalV2N by interactive exploration of the Ca^{2+} imaging window and Ca^{2+} trace window. A cell becomes verified after a visual confirmation of its footprint, Ca^{2+} Image, and Ca^{2+} trace, ensuring identifiable Ca^{2+} transients are present (**Fig. S3**). Once verified, the cell can be repurposed for machine learning training.
- Rejected Cell: Cells can be assigned to the rejection column for different reasons, including no identifiable Ca^{2+} transients, suspicious footprint, statistically identified as an outlier of the detected Ca^{2+} transients, etc. CalTrig provides the option to provide a written justification for each cell rejection, which can be used to provide feedback and potentially rerun the CNMF pipeline.
- Missing Cell: Automatic time-series based algorithms may not be sensitive enough to detect cells with low to medium SNR, or those constitutively active or inactive with minimal dynamic changes. While reviewing the Ca^{2+} imaging video, one may notice a potentially missed cell by CalV2N. CalTrig allows user to manually draw the contour of the suspected cell in the Ca^{2+} image window, creating its footprint (**Fig. S4**). The corresponding temporal trace of signal intensity for the selected area is

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

then calculated by averaging pixel intensities and displayed in the Ca^{2+} trace window. If the Ca^{2+} traces meet the criteria for accepted cells, the manually identified cell can be added to the list of missing cells.

Evaluation of CaIV2N performance

Although many CaIV2N tools have been developed, there is no evaluation system available to assess the quality of cell identification and Ca^{2+} transient extraction. One of our initial motivations for developing CalTrig was to address this gap by creating a platform to evaluate CaIV2N outputs (**Fig. 3**). Cells identified by CaIV2N can be accepted or rejected, and missing cells can be manually added as described above. Further refinement of cell quality evaluation is possible after Ca^{2+} transient detection, using metrics such as rising time, peak amplitude, and inter-transient interval to review transient kinetics. If the rate of missing cells is too high (e.g., 10% or higher) and/or the rate of acceptable cells is too low (e.g., 90% or lower), further optimization of CaIV2N parameters should be considered. For example, MiniAn pipeline can be optimized by adjusting the following parameters[9].

ksize: Controls the denoising step, filtering out electronic noise while preserving important cell details.

wnd: Regulates the background removal step, subtracting unwanted light to isolate signals from cells.

dl_wnd: Used in the spatial update to focus on nearby cells and adjust the shape of cell signals.

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

sparse_penal: In the spatial update, it controls how detailed or simplified the shapes of the detected cells are. In the temporal update, it balances how much detail is included in the signals from the cells. A higher value makes the signals simpler and less detailed. The best value for this needs to be found through experimentation, as it's hard to estimate in advance.

sparsity: Ensures only the most meaningful signals are captured during the temporal update, minimizing noise.

Low SNR may also indicate poor raw data quality, which can be addressed by optimizing AAV preparation (e.g., selecting appropriate subtypes, adjusting titer and volume), improving surgical techniques (slower needle insertion / withdrawal, slower AAV delivery, minimizing bleeding), and enhancing data collection methods (adjusting focal levels, securing miniScope anchoring, preventing cable twists via sensitive commutator, and ensuring reliable hardware connections).

2.3.4. Ca^{2+} transient identification

Transient confirmation can be done in three ways, including manually selecting the start and end points of the transients, auto filtering with specified kinetic parameters or specifying frame number ranges, or running a trained machine learning model.

2.3.4.1. Manual Identification:

Method: Ca^{2+} transients can be directly identified by manually selecting the start and end points of the rising section of Ca^{2+} transients (**Fig. S5**). Efforts have been made to

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

improve the efficiency of manual identification. For example, the window size of the Ca^{2+} can be pre-set at a desired size (e.g., 1000 frames per window width in the x-axis, -1 to 20 as the signal range in the y-axis) which will allow the visual impression of Ca^{2+} at different time stage, different animals are comparable, assisting a consistent decision making on Ca^{2+} manual identification. We also set up the shortcut keys to switch the screen window in the Ca^{2+} trace panel. Specifically, clicking A and F allow to jump to the first or the last window, respectively, and clicking S and D allow to jump to the previous or the next screen window, respectively. To account for inaccuracies resulting from human interaction, the selected points would automatically position themselves within a 20-frame window to the local maxima or minima, dependent on their order.

Applications for manual identification: **First**, directly label the Ca^{2+} transient at the beginning stage of the project when no reference in setting up the parameters for autodetection or no data set are available to training the machine learning model. It takes ~5 min to go through a 15-min Ca^{2+} traces. **Second**, correct, add, or remove the transient spikes after running the parameter-based auto detection to establish the ground truth for training the machine learning model. This would primarily apply to problematic spikes around the specified parameter threshold or those whose rising part was too slow to be confidently identified as a transient event. It takes ~1 min to go through a 15 min Ca2+ traces. **Third**, to further improve the Ca^{2+} transient identification after processed by an established machine learning model. Since our machine learning model, specifically the GRU model used in identifying the Ca^{2+} transient in neurons detected during the same recording window from the same animal are highly reliable

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

(more details in **Results**), we expect this correction process should be done on 2-4 cells per min. **Fourth**, when we need to extend the application of a well-trained machine learning model to a different brain region, a different cell type, etc., the machine learning model may provide a compromised predictability in detecting Ca^{2+} transients. Adding a few traces with the manually identified Ca^{2+} transient will set up a new data set as ground truth for establishing an updated machine learning model to be used in a new task.

2.3.4.2. Parameter-based automatic identification

Three Parameters

"Peak Threshold (ΔF/F)": First, let's define ΔF/F, which represents the relative change in fluorescence intensity, where:

- **ΔF** is the difference between the observed fluorescence intensity (**F**) at a given time and the background fluorescence intensity (**F_0**).
- **F_0** is the background fluorescence intensity, which we've calculated using a moving percentile provided by Caiman[4].

The formula is:

$$\Delta F/F = \frac{F - F_0}{F_0}$$

The peak ΔF/F value within a specified time window is considered as the potential peak of the Ca^{2+} transient. The "Peak Threshold (ΔF/F)" is the minimum value of peak ΔF/F. A Ca^{2+} transient is accepted only if its peak surpasses this threshold.

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

“Interval Threshold”: This specified the minimum Inter-Transient Interval (ITI), measured as the frame distance between the initial fames of two adjacent Ca^{2+} transients. If the distance between two transients is shorter than the “Interval Threshold”, they are considered part of the same transients, with the lower peak, whether preceding or following, merged into the higher peak.

“SNR Threshold”: This sets the minimal Signal-to-Noise ratio (SNR). Using the Savitzky–Golay filter [10,11] to smooth ΔF/F signals, noise is calculated as the difference between the original and filtered ΔF/F, further smoothed by a rolling window strategy. Then SNR is computed by dividing the smoothed ΔF/F by the estimated noise. See more details in **Supplementary Information**.

At the initial experimental stage, parameter settings are determined using manual identification on a limited number of cells or demo data. As more cells are manually identified, these parameters can be refined to enhance the effectiveness of parameter-based auto-detection.

Method: Through manual identification, the C signal from CNMF was found to reliably predict Ca^{2+} transients in most cases but prone to false positives. To enhance detection accuracy, the parameter-based auto detection algorithm starts by including all C picks in a candidate pool, then filtering down to a valid subset based on three pre-defined parameters: “Peak Threshold (ΔF/F)”, “Interval Threshold”, and “SNR Threshold” (**Fig. S7**). This is achieved through the following steps**.**

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

- Initial peak detection through local maxima calculation of the C signal.
- Refine peak selection from left to right by looping the following steps.
- Remove erroneous peaks if the S signal is zero.
- Allocate overlapping, continuous S signal to the current peak.
- Check if the distance to the next peak is shorter than the pre-defined "Interval Threshold".
- If so, merge the current and subsequent peaks, selecting the taller peak.
- If not, use the S signal to determine the start and end of the transient.
  - Accept the transient selection if in the defined boundaries, the values surpass both the pre-defined "Peak Threshold (ΔF/F)" and "SNR Threshold".

Application: Parameter based Ca^{2+} transient detection is faster than manual detection, but less predictive than using a well-trained machine learning model. It is particularly useful after manual identification of a few cells, enabling parameter tuning. Parameter-based detection can then serve as a pre-detection step to accelerate further manual identification, eventually creating a ground-truth dataset for training machine learning models. This method also helps refine or update machine learning models for different experimental conditions (**Figs. 6, 8,9**).

2.3.4.3. Detection using Machine Learning Model

Feature selection for machine learning model**:**

In selecting the appropriate architecture for the machine learning model, we aimed to incorporate key signals gained from the "ground truth" Ca^{2+} transient verification by

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

manual detection process. We identified three signals, i.e., C and ΔF/F, as sufficient for decision-making. The determination of whether a given time-step contains a Ca^{2+} transient can be inferred from a 100-frame window centered by the frame of interest. A corresponding data set, denoted E, was created by marking the initial and peaking frames within the context of a dynamic rise in Ca^{2+} transients.

Selection of machine learning models:

Ca^{2+} transient data are inherently time-series data with a long sequence of many neurons firing at different times, where the signal at any given frame is highly relevant to past and upcoming activity. Thus, Recurrent Neural Networks (RNNs) [12] and Transformers[13,14] are selected as the candidate machine learning models for Ca^{2+} transient detection due to their ability or potential to handle the temporal dynamics of Ca^{2+} signals in neurons.

The core of its architecture lies in the RNN cells and its gating mechanisms, which processes inputs sequentially, creating a representation of the data in the context of the preceding timestamps. In the case of Ca^{2+} transient data, two features (i.e., C, ΔF/F) used for training will be incorporated into the hidden state (*H*) and passed to the subsequent cell (*C*). We tested two RNN variants: Long Short-Term Memory (LSTM) [15,16] and Gated Recurrent Unit (GRU)[17,18]. Given the importance of both forward and backward context in detecting Ca^{2+} transients, we employed a bidirectional RNN. The outputs from both directions are concatenated and passed through a feed-forward layer to generate the final prediction.

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

One significant limitation of the RNN is due to the fixed size of its hidden state and the sequential nature of the architecture, causing early-time step bias. To address this, we trained a Transformer, which uses self-attention to compare the relationship between all input values simultaneously, rather than processing sequentially[19,20]. This approach allows the model to attend to both low and high-intensity values that occur at different temporal distances from the Ca^{2+} transient, improving prediction accuracy. Recent advancements in Natural Language Processing have introduced the **Local Transformer**, which limits attention to a subset of nearby inputs [21,22]. This model could help reduce errors by focusing on local context and ignoring distant irrelevant signals. However, the self-attention mechanism in the Transformer obscures positional information, hence we need positional embeddings. The trainable embeddings were used in our Transformer. Additionally, we observed that the small dimensionality of the original input data negatively affected training, likely due to the model's inability to effectively encode positional information in low-dimensional data. To address this, we introduced a preliminary dimension expansion layer, which projects the input into a higher-dimensional space before passing it through the encoding layers.

Both RNN and Transformer architectures were implemented in CalTrig using PyTorch [23,24]. For the Transformer, we used the Transformer Encoder module, and for the Local Transformer, we adapted a modified version of the model [25]. Parameters for machine learning model training are listed in **Table 2**.

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Table 2. Parameters for machine learning model training

Parameters	GRU & LSTM	Transformer	local Transformer
Hidden Ssize	30	42	32
Number of layers	3	3	3
Loss function	BCEWithLogitsLoss	BCEWithLogitsLoss	BCEWithLogitsLoss
Optimizer	Adam	Adam	Adam
Initial learning rate	0.001	0.001	0.001
Section Length	200	200	200
Slack	50	50	50
number of Heads		2	1
Local Attention Window Size			10
Look forward/backward Size			5 Local Attention Windows

Data Pre-processing:

The input data consisted of two signals (i.e., C, ΔF/F) representing 15-minute time intervals of 27000 frames per cell, captured at a sampling rate of 30 Hz. Each segment was normalized relative to the highest value within the cell, per data type. We tested two data generation and loading approaches:

Discrete Sample Chunking: We defined three variables—sequence length, slack, and rolling parameter. The sequence length specifies the number of frames the model will train on and use to make predictions, typically set to around 200 frames in our testing. The slack variable determines the extra context length on either side of a given sequence, providing necessary context for making predictions without being used for the predictions themselves. This slack is usually set between 50 and 100 frames. For sequences at the edges of the 27,000-frame segment, zero-padding is applied to match the slack length and maintain consistency. Lastly, the rolling parameter defines the

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

windowing approach for generating data, allowing overlapping sequences to be extracted for model training.

Ca^{2+} transient events account for only 2-3% of the overall data. To avoid biasing the model towards detecting non-Ca^{2+} transients, we initially applied class weighting, which resulted in a poor precision score. We investigated that it was due to the underemphasizing of noisy data within training, whose signal characteristics were more similar to transient activity rather than an empty signal. This resulted in a model that considered any activity including noise to be a transient event. We opted instead to implement stratification where we ensured that only samples containing ground-truth transient events or positive values from the C array were included. The C array was used as a reference to give the model insight into problematic spikes identified by the CNMF process but deemed invalid by the verifier. During classification, we average all outputs for a single time-step to address overlapping predictions.

Key metrics for validation:

We used Precision, Recall, F1 and macro F1 as key metrics to evaluate the performance of machine learning model in predicting Ca^{2+} transients vs. nonCa2+ transients (**Fig. S8**). There are four types of predictions: true positive (TP) predicting the positive as positive, true negative (TN) predicting the negative as negative, false positive (FP) predicting the negative as positive, and false (FN) negative predicting the positive as negative.

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Table 2. Event to be predicted: Ca^{2+} transients

	predicted Ca^{2+} transients	predicted non-Ca^{2+} transients
Actual Ca^{2+} transients	TP	FN
Actual non-Ca^{2+} transients	FP	TN

Table 3. Event to be predicted: non-Ca^{2+} transients

	predicted Ca^{2+} transients	predicted non-Ca^{2+} transients
Actual Ca^{2+} transients	TN	FP
Actual non-Ca^{2+} transients	FN	TP

Precision measures how many of the positive predictions (TP + FP) are actually positive (TP). (It focuses on the quality of the positive predictions.)

$$Precision = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{FP}}$$

Recall measures how many of the actual positive instances (TP + FN) are correctly predicted as positive (TP). (It focuses on the ability to find all positive cases.)

$$Recall = \frac{\mathrm{TP}}{\mathrm{TP} + \mathrm{NF}}$$

F1 score is the harmonic mean of Precision and Recall, balancing both metrics.

$$F1 = 2 \times \frac{\mathrm{Precision}\ \times \mathrm{Recall}}{\mathrm{Precision} + \mathrm{Recall}}$$

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Macro F1 is the average of the F1 scores calculated for each class, i.e., Ca^{2+} transients and non-Ca^{2+} transients.

$$macro\ F1 = \frac{\mathrm{Ca^{2+}\ transient\ F1 + non\text{-}Ca^{2+}\ Transient\ F1}}{2}$$

For evaluation, cells were randomly separated into training (80%), validation (10%), and testing (10%) sets (**Fig. 6**). Manual identification, parameter-based identification and machine learning model-based identification can be integrated in detecting Ca^{2+} (**Fig. 4**).

2.3.5. Data export

All the detected Ca^{2+} transient information can be extracted. A few examples are listed below. This will directly aid in data processing, statistical analyses, and figure preparation for sharing and publishing.

Animal-wide data export

A data table is generated, with each row representing one of the CalS2N-detected cells and each column providing specific readout as a general description of that cell (**Fig. S9A**). From left to right, the columns include: Cell ID, Cell Size (number of pixels), Footprint Location (x, y), Total Ca^{2+} Transient Count, Frequency (Hz), Average Amplitude (ΔF/F), Average Rising (# of frames), Average Rising Time (seconds), Average Interval (seconds), Standard Deviation (denoted Std, ΔF/F), Mean Absolute Deviation (denoted MAD, ΔF/F), Average Peak Amplitude (ΔF/F), and Category (Verified, Rejected, Missing). The data can be easily copied to the clipboard for further

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

processing in other applications like Excel (Microsoft 365) or Prism (GraphPad). Figures, such as box plots showing the 25%, 50%, and 75% values for metrics like Average Peak Amplitude or ITI (**Fig. S9B, C**), can be directly generated and saved as editable SVG files for publication purposes.

Cell-wide data export: A data table is generated for a selected cell, with the top row displaying column titles and the subsequent rows representing individual Ca^{2+} transients detected by CalTrig, one transient per row (**Fig. S10**). The columns include: Rising Start (frames), Rising Stop (frames), Total Rising Frames, Rising Start (seconds), Rising Stop (seconds), Total Rising Time (seconds), Interval with Previous Transient (frames), Peak Amplitude (ΔF/F), and Total Amplitude (ΔF/F, calculated as the area under the rising slope). The column title can be selected to sort the data based the information in the selected column (**Fig. S10B**). This table can be copied to the clipboard for further processing in applications like Excel (Microsoft 365) or Prism (GraphPad). Figures such as Amplitude Distribution or ITI Frequency Histograms (**Fig. S10 C, D**) can be created directly from this data and saved as editable SVG files for publication.

Maximum projection image

An editable image file, including the footprints of detected or verified cells for an animal, can be created for data sharing and publication purposes.

2.4. Statistical Analysis:

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Data were collected from 6 mice *in vivo,* shown as mean ± SEM in curve graphs (**Figs. 8B**, **C**, **F**, **G**, **J**, **K**, **N; 9B**, **C**, **F**, **G**, **J**, **K**, **N**) or the quartile in violin plots (**Figs. 6**; **7**; **8D**, **E**, **H**, **I**, **L**, **M**, **O**; **8D**, **E**, **H**, **I**, **L**, **M**, **O**). Using GraphPad Prism 10, statistical significance was assessed by one way ANOVA or two-way ANOVA, followed by Bonferroni post-hoc tests. Statistical significance was considered to be achieved if $p < 0.05$.

3. RESULTS:

3.1. Comparison of the 4 machine learning models

Our goal is to find an efficient machine learning model with high predictability in detecting the Ca^{2+} transients while minimizing computational demands. To achieve this, we compared the performance of two RNN variants, GRU and LSTM, alongside two Transformer models, the standard Transformer and local Transformer. We used a dataset of 203 cells with manually confirmed Ca^{2+} transient as “ground” truth (see more details in Methods) (**Fig. 6A**). Our data showed that the macro F1 score for predicting Ca^{2+} transients and no Ca^{2+} transients was higher with RNN (GRU, 0.948 ± 0.002; LSTM, 0.946±0.004) compared to the standard Transformer (0.919 ± 0.005), which was partially improved by local Transformer (0.935 ± 0.005) (**Fig. 6H**).

For Ca^{2+} transient prediction, the F1 score is higher with RNN (GRU, 0.900 ± 0.005; LSTM, 0.894±0.008) compared to the standard Transformer (0.841 ± 0.010), which was partially improved by local Transformer (0.873 ± 0.010) (**Fig. 6F**). In terms of Precision, both GRU (0.873 ± 0.008) and LSTM (0.860 ± 0.010) models outperformed the standard Transformer (0.771 ± 0.015) and the local Transformer (0.835 ± 0.013) (**Fig.**

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

6B). However, the Recall scores across 4 machine learning models were similar (GRU: 0.926 ± 0.008; LSTM: 0.931 ± 0.010; standard Transformer: 0.929 ± 0.009; local Transformer: 0.916 ± 0.010) (**Fig. 6D**), indicating that the F1 score differences were mainly driven by Precision.

For predicting non-Ca^{2+} transients, the precision, recall, and F1 scores were all near perfect (>0.990), due to the consistent nature of non-Ca^{2+} transient signals. When analyzing F1 scores specifically for non-Ca^{2+} transient prediction, similar trends as the macro F1 scores were observed across the four MACHINE LEARNING models (**Fig. 6G**), with differences primarily driven by Recall (**Fig. 6E**) rather than Precision (**Fig. 6C**).

Compared to Transformer models, RNN models exhibited higher Precision in detecting Ca^{2+} transients and better Recall for non-Ca^{2+} transients, both attributable to RNN's ability to minimize misclassification of the non-Ca^{2+} transient signal as Ca^{2+} transient signals. The consistent good Recall for Ca^{2+} transients and Precision values for non Ca^{2+} transients across all MACHINE LEARNING models indicates that they have a low likelihood of misclassifying Ca^{2+} transients as non-Ca^{2+} transients. The primary challenge in identifying Ca^{2+} transients lies in reducing errors where non-Ca^{2+} transients are mistakenly classified as transients. RNN models outperformed Transformer in addressing this issue, which was partially improved when using the local Transformer. Given its lower variability and faster processing time, GRU is recommended over LSTM. Therefore, all subsequent analyses were conducted using the GRU model.

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

3.2. Optimization of the cell number of the training data set

To determine the optimal number of cells to be used as the training dataset, we tested the predictability of GRU model trained by varying number of cells, ranging from 1 to 20. When the testing data sets were randomly sampled form the same session stage (i.e., the first 15 min or the last 15 min) in the same 1hr daily session on either Day 1 or Day 5, we observed a significant improvement in the macro F1 score when 5 to 20 cells were sampled, compared to GRU models trained with only 1 or 2 cells. However, no differences were observed among models trained with 5 to 20 cells (mean ± S.D.E macro F1 score: 1 cell, 0.817 ± 0.014; 2 cells, 0.885 ± 0.008; 5 cells, 0.926 ± 0.005; 10 cells, 0.945 ± 0.004; 15 cells, 0.947 ± 0.004; 20 cells, 0.946 ± 0.004) (**Fig. 7H**). This cell number-dependent predictability is primarily attributable to the model's predictability of the Ca^{2+} transient indicated by its Precision (**Fig. 7B**), Recall (**Fig. 7D**), and F1 scores (**Fig. 7F**) for Ca^{2+} transients, as the Precision (**Fig. 7C**), Recall (**Fig. 7E**) and F1 scores (**Fig. 7G**) for non-Ca^{2+} transient were all near perfect regardless of the cell number of the training data set. We further explored the effects of cell number when testing cells were randomly sampled from the different session stages within the same 1-hr daily session, the same session stage on a different daily session, and the different sessions stage on a different daily session of the same mouse (**Fig. 8A**). In call cases, we found significant improvements of both Ca^{2+} transient prediction (**Fig**. **8B**, **F**, **J**, **N**) and the non-Ca^{2+} transient predictions – though the latter were near perfect in most instances (**Fig**. **8 C**, **G**, **K**, **O**)– when 5 or more cells were sampled for training, compared to the models trained with only 1 or 2 cells. Finally, we extended further by randomly sampling the training data sets from different mice (**Fig. 9A**). We found significant improvements

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

of Ca^{2+} transient prediction (**Fig**. **9 B**, **F**, **J**, **N**) and the non-Ca^{2+} transient prediction - though the latter were near perfect in most instances (**Fig**. **9 C**, **G**, **K**, **O**)- when using 5 or more cells, regardless of whether the testing cells were from the same or different brain regions in mice distinct from those used for training. In conclusion, the number of cells used for training is crucial for accurately predicting Ca^{2+} transients. A minimum of 10 cells appears to be sufficient for achieving macro F1 scores above 0.900 when training the GRU model.

3.3. Share the machine learning model for predicting testing cells across different time windows in the same mouse

To assess how well a GRU model trained on data from a section of a 1-hr daily session can predict Ca^{2+} transients from any other recordings in the same mouse, we randomly sampled testing cells under four conditions: (1) the same session stage in the same daily session (smSS::smDay), (2) a different session stage in the same daily session (dfSS::smDay), (3) the same session stage in a different daily session (smSS::dfDay), and (4) a different session stage in a different daily session (dfSS::dfDay) (**Fig. 8A**). When 20 cells were sampled, the macro F1 scores were 0.946 ± 0.004 for smSS::smDay, 0.936 ± 0.006 for dfSS::smDay, 0.934 ± 0.005 for smSS::dfDay, and 0.927 ± 0.008 for dfSS::dfDay (**Fig. 8N**). Thus overall, the predictability remained high when testing cells were sampled from the same mouse used for GRU model training, though the macro F1 score for predicting both Ca^{2+} transients and non-Ca^{2+} transients was affected statistically, but mildly, by the source of testing cells (**Fig. 8 N, O**).

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

For predicting Ca^{2+} transients, the source of testing cells affected the F1 scores (**Fig. 8 J, L**), particularly the Precision (**Fig. 8 B, D**) rather than the Recall (**Fig. 8 F, H**). When smSS::smDay was set as the gold standard with 20 cells as the benchmark, similar predictability was observed when testing cells were sampled from dfSS::smDay or smSS::dfDay. However, lower Precision and F1 score, but comparable Recall, were noted when testing cells were sampled from dfSS::dfDay.

For predicting non-Ca^{2+} transients, the source also affected the F1 scores (**Fig. 8 K, M**), primarily impacting the Recall (**Fig. 8 G, I**), while the Precision remained largely unaffected (**Fig. 8 C, E**). Similarly, as what was observed for predicting Ca^{2+} transients, we found the non-Ca^{2+} transient prediction were comparable when testing cells were sampled from smSS::smDay, dfSS::smDay or smSS::dfDay. Lower Recall and F1 score, but similar Precision, for non-Ca^{2+} transient prediction was detected when testing cells were sampled from dfSS::dfDay.

In conclusion, the GRU model can effectively predict Ca^{2+} transients in cells from the same mouse used for training. However, the risk of misclassifying non-Ca^{2+} transients as Ca^{2+} transients increase, though within a narrow range, when testing cells are sampled from a different session stage and a different day.

3.4. Share the machine learning model for predicting testing cells from different mice

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

To evaluate how the GRU model can predict the Ca^{2+} transients from a different mouse, either sharing the same brain region or switch to a different brain region, we use the smSS::smDay sampling strategy as the benchmark and added two more strategies in sampling testing cells: (1) the same brain region in a different mouse (denoted smRegion::dfMouse, and (2) a different brain region in a different mouse (denoted dfRegion::dfMouse). When 20 cells were sampled, the macro F1 scores were 0.946 ± 0.004 for smDay::smSS, 0.947 ± 0.003 for smRegion::dfMouse, and 0.927 ± 0.005 dfRegion::dfMouse. Thus, regardless of the brain region specificity, overall predictability remained high when testing cells were sampled from mice different from the one used for GRU training, although the macro F1 score for predicting both Ca2+ and non-Ca^{2+} transients was statistically reduced when the testing cells were sampled in dfRegion::dfMouse (**Fig. 9 N, O**).

For predicting Ca^{2+} transients in different mice, the source of testing cells affected the F1 scores (**Fig. 9 J, L**) by impacting both the Precision (**Fig. 9 B, D**) and the Recall (**Fig. 9 F, H**). Lower Precision, Recall and F1 score were noted when testing cells were sampled from dfRegion::dfMouse, relative to either smSS::smDay or dfRegion::dfMouse.

For predicting non-Ca^{2+} transients, the source also affected the F1 scores (**Fig. 9 K, M**), by impacting both the Precision (**Fig. 8 C, E**) and the Recall (**Fig. 8 G, I**). Lower, although still near perfect, Precision, Recall and F1 score were noted when testing cells

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

were sampled from dfRegion::dfMouse, relative to either smSS::smDay or dfRegion::dfMouse.

In conclusion, the GRU model can identify Ca^{2+} transients in cells from different mice with a macro F1 beyond 0.900. However, the risk of misclassifying non-Ca^{2+} transients as Ca^{2+} transients or Ca^{2+} transients as non-Ca^{2+} transients increase, though within a narrow range, when testing cells are sampled from a different brain region of a different mouse. There appears to be no significant impact on the F1 score when the testing cells are from the same region of a different mouse.

4. DISCUSSION

4.1. Selection of Machine Learning Model

In recent years, there a few studies using different machine learning methods which have been developed that have utilized some combination of the CNN, Attention-based architectures for cell identification and subsequent cell extraction[26-28]. However, they require a significant amount of labelled data as ground truth to achieve adequate predictive performance. This raises concern in both time investment to train a single model as well as due to the black box nature of these models, their ability to maintain performance given a substantial change in the input data (such as changes to different experimental animal, another rain region of interested, a different type of neurons).

In this study, we evaluated four machine learning models (GRU, LSTM, Transformer, and Local Transformer) for Ca^{2+} transient detection. Both predictability performance

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

(i.e., precision, recall, and F1 scores) and time efficiency should be considered. **First**, GRU model is the most time-efficient, with a computation time of 287.1 second / epoch, slightly faster than LSTM (290.59 second / epoch) due to its simpler architecture and fewer parameters, making it computationally less expensive. GRU also exhibits strong overall performance in terms of precision, recall, and F1 scores, making it an excellent choice when considering both predictability and computation time. **Second**, LSTM takes marginally longer than GRU but performs slightly worse across multiple predictability metrics for Ca^{2+} transient detection. While very close to GRU in time efficiency, its more complex gating mechanism results in slightly longer computation times, without a significant performance advantage. **Third**, standard Transformer requires significantly more computation time at 708.8 second / epoch. Although transformers can be easily parallelized, their global self-attention mechanism and large parameter space demand much higher computational power. This increased time is not justified by any improvement in Ca^{2+} transient predictability, which actually suffers due to interruptions from distal frames when global attention is used in decision-making. **Fourth**, Local Transformer is the least time-efficient, taking 1744.4 second / epoch, almost 6x slower than GRU or LSTM, and 2.5x slower than the standard Transformer. This can be attributed to the increased complexity of managing localized attention windows and maintaining context between them. Additionally, it loses parallelization efficiency when processing the entire window. However, this increased computation time is compensated by improved predictability, as attention is focused more effectively on local frames, leading to better decision-making. **In conclusion**, RNNs outperform transformers in both time efficiency and predictability for Ca^{2+} transient detection.

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Between two RNN variants, GRU offers a better performance in both time efficiency and predictability, although LSTM remains a viable alternative. When the GRU performance is challenged by dataset size, computational resources, and the complexity of the transient dynamics, other models could be explored directly in the CalTrig.

4.2. Integrative Visual Exploration

The synchronization of Ca^{2+} imaging, cellular footprint, behavioral tracking, and Ca^{2+} trace windows, adjustable via a time bar, allows for a comprehensive and temporally aligned visualization of neuronal activity and behavior. This integration of multiple data streams provides several key benefits for neuroscience research. **First**, enhanced understanding of brain-behavior relationships: By synchronizing Ca^{2+} traces with behavioral data, researchers can directly observe how specific neuronal populations respond during behaviors or external stimuli. This real-time insight deepens our understanding of how brain activity drives behavior, such as the timing of Ca^{2+} transients relative to actions like lever presses. **Second**, technical flexibility and real-time adjustments: The adjustable time window enables researchers to zoom in on specific segments of data, closely examining how neural activity changes just before or after a behavioral event, which is critical for understanding causal relationships between stimuli and neural responses. **Third**, hypothesis proposing and testing: The integration of data streams facilitates hypothesis proposing, which can be rapidly tested about the relationships between neural signals and behaviors. Researchers can observe whether certain Ca^{2+} transients precede or follow behavioral events, helping to refine hypotheses in real-time. **Fourth**, visualization of micro-network dynamics: Viewing

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

neural footprints while tracking behavior enables the identification of micro-networks at the neuronal level, distinct from traditional brain region-level network, during specific tasks or stimuli. This is critical for understanding neuronal network dynamics in healthy and diseased states, such as in learning, memory, or addiction studies. **Fifth**, categorization of neurons: Visualizing Ca^{2+} transients allows for the clustering of neurons based on activity patterns, helping to uncover subpopulations with distinct response profiles. This helps researchers understand the heterogeneity and complexity of neural networks and their roles in both normal and pathological behaviors. **Sixth**, facilitating longitudinal studies: In long-term studies, where animals undergo repeated trials, integrative visualization tools allow researchers to track changes in both neuronal activity and behavior over time, aiding in the study of neuroplasticity and how neuronal responses evolve with experience. **Seventh**, improved data interpretation and collaboration: The ability to visualize multiple layers of data simultaneously aids in the interpretation of complex datasets. This holistic view helps researchers identify patterns and facilitates collaboration by making findings more accessible and easier to communicate across disciplines. **In conclusion**, integrative visual exploration enables deeper exploration of brain function, supports hypothesis generation about neuronal mechanisms, and opens new avenues for understanding neurological disorders and developing treatments.

4.3. Featured Value of CalTrig

The **usability** of the CalTrig tool is highlighted by its integrated visual experience, which provides users with a comprehensive view of Ca^{2+} images, neuron footprints, and Ca^{2+}

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

transient traces in one unified interface. The tool features multi-layered toolboxes that cater to various analytical needs, allowing users to work efficiently at both the cellular and Ca^{2+} transient levels. Manual identification of Ca^{2+} transients is made efficient with features such as keyboard shortcuts (ASDF keys), hide/show traces, and unified scales for transient identification. This allows users to manually identify transients in approximately 1-2 minutes per 15-minute Ca^{2+} trace. Furthermore, manual identification can be assisted by auto-identification processes based on kinetic parameters or pre-established machine learning models. CalTrig is a self-looped tool, designed for continuous functional improvement, ensuring it remains adaptable and upgradable.

In terms of **accessibility**, CalTrig is a GUI-based open-source tool, making it readily shareable and open for further updates. It supports all CNMF or CNMF-E-processed imaging data and operates with limited computing demands, allowing for efficient performance even on less powerful systems. The graphical user interface (GUI) of CalTrig is designed to be user-friendly, ensuring that non-programmers can easily interact with it through intuitive buttons, menus, and forms. The tool presents a professional appearance, which enhances its suitability for presentations, collaborations, and publications. Real-time feedback is provided during parameter adjustments in Ca^{2+} transient identification, allowing users to fine-tune the process interactively. CalTrig can be downloaded as an independent application that runs in a Python environment, meaning users do not need additional software installations. Additionally, the tool is highly customizable, allowing users to tailor it to specific

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

workflows and improving productivity for different use cases. Its expandability ensures that it can accommodate future updates and feature integrations.

Our research demonstrates that the GRU model provides high predictability when applied to testing cells from the same or different session stages, across various training days, brain regions, and mice. The GRU model has proven to be efficient in training, as shown in Fig. 7, even with a limited number of cells (Figs. 8-10), and the "ground truth" of Ca^{2+} transients can be established with relative ease, as detailed in the Methods and Results sections. This makes it highly feasible to extend testing to longer training sessions, such as the 6-hour sessions commonly used in studies on learning, memory, motivation, and addiction, or across broader time spans covering days, weeks, or months. The tool we developed here will be instrumental in evaluating the feasibility of *in vivo* Ca^{2+} recordings during extended sessions over prolonged recording periods.

The high predictability of the GRU model in detecting Ca^{2+} transients also indicates that the basic properties of Ca^{2+} transients, such as rise slope, amplitude, and signal-to-noise ratio, remained stable during 1-hour recording sessions, across five recording days, and across different brain regions in several mice. Notably, the neurons detected in two brain regions, M2 and PrL, are most likely pyramidal neurons, which may explain the model's high level of expandability. A future direction for research could be to test the expandability of the model to different types of neurons.

4.4. Limitations and Future Directions

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

There are several areas in which the current study has not yet explored. For instance, we have not tested the trained machine learning model for detecting Ca^{2+} transients across different animal lines or across species, such as from mice to rats. The Ca^{2+} indicator used in this study is GCaMP8f, and it would be interesting to compare the dynamics of Ca^{2+} transients and the persistence of fluorescent intensity using other genetically encoded Ca^{2+} indicators (GECIs). Furthermore, the detection of Ca^{2+} transients is often not the end goal of data analysis. When conducting longitudinal recordings across different experimental groups, researchers may face challenges related to selecting appropriate time windows for analysis. Behavior-associated Ca^{2+} transients present a particularly valuable area for exploration, as they offer insights into neuronal activities linked to specific brain functions. Another area of interest is the association between neuronal footprints and activity in specific physiological or pathological conditions. This is a largely unexplored field, but it is intriguing to consider how the spatial distribution of neurons may influence brain function. Our team is currently developing relevant tools to integrate with CalTrig, which will further enhance our ability to understand the role of Ca^{2+} transients in brain output coding.

ACKNOWLEDGEMENTS

This work was supported by NIH grants (R01AG072897, R01AA025784, and R01DA059548). We thank the support from High-Performance Computing and Storage at Indiana University. This work also received support from the Indiana Spinal Cord & Brain Injury Research Fund (Indiana State Department of Health); its contents are solely the responsibility of the providers and do not necessarily represent the official views of

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

the Indiana State Department of Health. We acknowledge Dr. Denise Cai and her team for developing MiniAn as a powerful tool used to extract Ca^{2+} traces.

AUTHOR CONTRIBUTION

Experimental design: MAL, YC, HF, CG, YYM. Data collection: MAL, YC. Data analysis: MAL, YC, AK, CG, YM. Result discussion: MAL, YC, HF, AK, CG, YYM. Manuscript preparation: MAL, YC, HF, AK, CG, YYM.

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

REFERENCE

1. Pachitariu, M., Stringer, C., Dipoppa, M., Schröder, S., Rossi, L.F., Dalgleish, H., Carandini, M., and Harris, K.D. (2016). Suite2p: beyond 10,000 neurons with standard two-photon microscopy. bioRxiv.
2. Kaifosh, P., Zaremba, J.D., Danielson, N.B., and Losonczy, A. (2014). SIMA: Python software for analysis of dynamic fluorescence imaging data. Front Neuroinform *8*, 80. 10.3389/fninf.2014.00080.
3. Soltanian-Zadeh, S., Sahingur, K., Blau, S., Gong, Y., and Farsiu, S. (2019). Fast and robust active neuron segmentation in two-photon calcium imaging using spatiotemporal deep learning. Proc Natl Acad Sci U S A *116*, 8554-8563. 10.1073/pnas.1812995116.
4. Giovannucci, A., Friedrich, J., Gunn, P., Kalfon, J., Brown, B.L., Koay, S.A., Taxidis, J., Najafi, F., Gauthier, J.L., Zhou, P., et al. (2019). CaImAn an open source tool for scalable calcium imaging data analysis. Elife *8*. 10.7554/eLife.38173.
5. Lu, J., Li, C., Singh-Alvarado, J., Zhou, Z.C., Frohlich, F., Mooney, R., and Wang, F. (2018). MIN1PIPE: A Miniscope 1-Photon-Based Calcium Imaging Signal Extraction Pipeline. Cell Rep *23*, 3673-3684. 10.1016/j.celrep.2018.05.062.
6. Zhou, P., Resendez, S.L., Rodriguez-Romaguera, J., Jimenez, J.C., Neufeld, S.Q., Giovannucci, A., Friedrich, J., Pnevmatikakis, E.A., Stuber, G.D., Hen, R., et al. (2018). Efficient and accurate extraction of in vivo calcium signals from microendoscopic video data. Elife *7*. 10.7554/eLife.28728.
7. Mukamel, E.A., Nimmerjahn, A., and Schnitzer, M.J. (2009). Automated analysis of cellular signals from large-scale calcium imaging data. Neuron *63*, 747-760. 10.1016/j.neuron.2009.08.009.
8. Huang, D., and Ma, Y.Y. (2023). Increased Excitability of Layer 2 Cortical Pyramidal Neurons in the Supplementary Motor Cortex Underlies High Cocaine-Seeking Behaviors. Biol Psychiatry. 10.1016/j.biopsych.2023.06.002.
9. Dong, Z., Mau, W., Feng, Y., Pennington, Z., Chen, L., Zaki, Y., Rajan, K., Shuman, T., Aharoni, D., and Cai, D.J. (2021). Minian: An open-source miniscope analysis pipeline. bioRxiv. https://doi.org/10.1101/2021.05.03.442492.
10. Steinier, J., Termonia, Y., and Deltour, J. (1972). Smoothing and differentiation of data by simplified least square procedure. Anal Chem *44*, 1906-1909. 10.1021/ac60319a045.
11. Dai, W., Selesnick, I., Rizzo, J.R., Rucker, J., and Hudson, T. (2017). A nonlinear generalization of the Savitzky-Golay filter and the quantitative analysis of saccades. J Vis *17*, 10. 10.1167/17.9.10.
12. Durstewitz, D., Koppe, G., and Thurm, M.I. (2023). Reconstructing computational system dynamics from neural data with recurrent neural networks. Nat Rev Neurosci *24*, 693-710. 10.1038/s41583-023-00740-7.
13. Bian, Q., As'arry, A., Cong, X., Rezali, K., and Raja Ahmad, R. (2024). A hybrid Transformer-LSTM model apply to glucose prediction. PLoS One *19*, e0310084. 10.1371/journal.pone.0310084.
14. Chandra, A., Tunnermann, L., Lofstedt, T., and Gratz, R. (2023). Transformer-based deep learning for predicting protein properties in the life sciences. Elife *12*. 10.7554/eLife.82819.
15. Jiang, F., Yuen, K.K.R., and Lee, E.W.M. (2020). A long short-term memory-based framework for crash detection on freeways with traffic data of different temporal resolutions. Accid Anal Prev *141*, 105520. 10.1016/j.aap.2020.105520.
16. Alfattni, G., Peek, N., and Nenadic, G. (2021). Attention-based bidirectional long short-term memory networks for extracting temporal relationships from clinical discharge summaries. J Biomed Inform *123*, 103915. 10.1016/j.jbi.2021.103915.
17. Merkelbach, K., Schaper, S., Diedrich, C., Fritsch, S.J., and Schuppert, A. (2023). Novel architecture for gated recurrent unit autoencoder trained on time series from electronic health

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

records enables detection of ICU patient subgroups. Sci Rep *13*, 4053. 10.1038/s41598-023-30986-1.

18. Wang, R., Li, C., Fu, W., and Tang, G. (2020). Deep Learning Method Based on Gated Recurrent Unit and Variational Mode Decomposition for Short-Term Wind Power Interval Prediction. IEEE Trans Neural Netw Learn Syst *31*, 3814-3827. 10.1109/TNNLS.2019.2946414.
19. Choi, S.R., and Lee, M. (2023). Transformer Architecture and Attention Mechanisms in Genome Data Analysis: A Comprehensive Review. Biology (Basel) *12*. 10.3390/biology12071033.
20. Liu, Z., Lv, Q., Yang, Z., Li, Y., Lee, C.H., and Shen, L. (2023). Recent progress in transformer-based medical image analysis. Comput Biol Med *164*, 107268. 10.1016/j.compbiomed.2023.107268.
21. Boffi, N.M., and Vanden-Eijnden, E. (2024). Deep learning probability flows and entropy production rates in active matter. Proc Natl Acad Sci U S A *121*, e2318106121. 10.1073/pnas.2318106121.
22. Zhao, X., Yang, T., Li, B., Yang, A., Yan, Y., and Jiao, C. (2024). DiffGAN: An adversarial diffusion model with local transformer for MRI reconstruction. Magn Reson Imaging *109*, 108-119. 10.1016/j.mri.2024.03.017.
23. Orken, M., Dina, O., Keylan, A., Tolganay, T., and Mohamed, O. (2022). A study of transformer-based end-to-end speech recognition system for Kazakh language. Sci Rep *12*, 8337. 10.1038/s41598-022-12260-y.
24. Lam, C., Thapa, R., Maharjan, J., Rahmani, K., Tso, C.F., Singh, N.P., Casie Chetty, S., and Mao, Q. (2022). Multitask Learning With Recurrent Neural Networks for Acute Respiratory Distress Syndrome Prediction Using Only Electronic Health Record Data: Model Development and Validation Study. JMIR Med Inform *10*, e36202. 10.2196/36202.
25. Wang, P. (2020). GitHub - lucidrains/local-attention: An implementation of local windowed attention for language modeling. GitHubhttps://github.com/lucidrains/local-attention.
26. Sita, L., Brondi, M., Lagomarsino de Leon Roig, P., Curreli, S., Panniello, M., Vecchia, D., and Fellin, T. (2022). A deep-learning approach for online cell identification and trace extraction in functional two-photon calcium imaging. Nat Commun *13*, 1529. 10.1038/s41467-022-29180-0.
27. Bao, Y., Soltanian-Zadeh, S., Farsiu, S., and Gong, Y. (2021). Segmentation of neurons from fluorescence calcium recordings beyond real time. Nature Machine Intelligence *3*, 2.
28. Denis, J., Dard, R.F., Quiroli, E., Cossart, R., and Picardo, M.A. (2020). DeepCINAC: A Deep-Learning-Based Python Toolbox for Inferring Calcium Imaging Neuronal Activity Based on Movie Visualization. eNeuro *7*. 10.1523/ENEURO.0038-20.2020.

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

FIGURE LEGENDS

Figure 1. Flowchart showing the procedures of an *in vivo* Ca^{2+} study.

Figure 2. Five windows in CalTrig interface

Figure 3. Use CalTrig to validate and filter cells identified by CalV2N

Figure 4. Use CalTrig to identify Ca^{2+} transients

Figure 5. Flowchart demonstrating Ca^{2+}transient validation via Machine Learning models.

A, Applications of RNN in predicting Ca^{2+} transients.

B, Applications of Transformer in predicting Ca^{2+} transients.

C, **D**, Diagrams showing the internal operations of two variants of RNN, i.e., GRU module (**C**) and LST module (**D**). The GRU uses two gates: the *update gate*, which controls the amount of information passed to the next step, and the *reset gate*, which determines how much of the previous information to forget. These gates modify the hidden state H(t) at the current time step based on the input X(t) and the previous hidden state H(t−1), leading to a new hidden state H(t). The LSTM module architecture is depicted, showcasing its cell structure. It uses three gates: *forget*, *input*, and *output* gates to control the flow of information. The input X(t), the previous cell state C(t−1), and the previous

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

hidden state H(t−1) are processed through these gates to update the cell state C(t) and produce the new hidden state H(t).

E, **F**, the diagram showing the Transform (**E**) and local Transformer (**F**) mechanisms. Each column corresponds to the same set of data. The non-translucent color in a column highlights the current input being evaluated, while translucent colors indicate the values being compared to it. No color signifies that certain values are excluded from evaluation for the current input.

FNN, Feedforward neural network.

Figure 6. GRU module has the best predictability of Ca^{2+} transients.

A, Shared strategy for assigning data to training, validation, and testing across four Machine learning models.

B, **C**, Precision varied significantly in Ca^{2+} transient prediction (**B**, $F_{3,36}$=14.2, p<0.01), but remained similar in no Ca^{2+} transient prediction (**C**, $F_{3,36}$=0.4, p=0.78) in four Machine Learning models.

D, **E**, Recall remained similar in Ca^{2+} transient prediction (**D**, $F_{3,36}$=0.6, p=0.63) but varied in Ca^{2+} transient prediction (**E**, $F_{3,36}$=9.4, p<0.01) in four Machine Learning models.

F, **G**, Significant differences in F1 scores of Ca^{2+} transient prediction (**F**, $F_{3,36}$=9.3, p<0.01) and no Ca^{2+} transient prediction (**G**, $F_{3,36}$=4.8, p<0.01) in four machine learning models.

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

H, Significant differences in macro F1 scores ($F_{3,36}$=9.3, p<0.01) in four Machine Learning models.

Each machine learning model was trained using data from 226 cells, with 28 cells allocated to the validation set and 28 cells to the testing set. Data were analyzed by one-way ANOVA, followed by Bonferroni *post hoc* test. *, p<0.05; **, p<0.01.

Figure 7. Prediction of Ca^{2+} transients or no Ca^{2+} transients across different numbers of cells in Machine learning training.

A, The strategy of randomly picking up cells for training, validation and testing within the same session stage on the same day.

B, **C**, Increasing the number of cells in machine learning training model significantly improved the precision in predicting Ca^{2+} transients (**B**, $F_{5,234}$=11.0, p<0.01) and no Ca^{2+} transients (**C**, $F_{5,234}$=12.8, p<0.01).

D, **E**, Increasing the number of cells in machine learning training model significantly improved the recall in predicting Ca^{2+} transients (**D**, $F_{5,234}$=40.9, p<0.01) and no Ca^{2+} transients (**E**, $F_{5,234}$=2.8, p=0.02).

F, **G**, Increasing the number of cells in machine learning training model significantly improved the F1 scores in predicting Ca^{2+} transients (**F**, $F_{5,234}$=46.7, p<0.01) and no Ca^{2+} transients (**G**, $F_{5,234}$=12.0, p<0.01).

H, Increasing the number of cells in machine learning training model significantly improved the macro F1 scores in predicting Ca^{2+} transients and no Ca^{2+} transients (**H**, $F_{5,234}$=46.6, p<0.01)

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Data were analyzed by one-way ANOVA, followed by Bonferroni *post hoc* test. *, $p<0.05$; **, $p<0.01$, compared to the machine learning model trained by 1 cell. $^{\#}$, $p<0.05$; $^{\#\#}$, 0.01, compared to the machine learning model trained by 2 cells. 40 testing cells in each group.

Figure 8.

A, Four strategies for sourcing cells within the M2 area from a mouse for machine learning model training, validation, and testing.

B-E, The training cell number, and the testing cell source, but not their interactions, significantly affected the Precision in predicting Ca^{2+} transients (**B**, training cell # $F_{5,936}=21.4$, $p<0.01$; testing cell source $F_{3,936}=10.0$, $p<0.01$; training cell number × testing cell source interaction $F_{15,936}=0.8$, $p=0.64$). The training cell number, but not the testing cell source or their interactions, affected the Precision in predicting no Ca^{2+} transients (**C**, training cell number $F_{5,936}=71.7$, $p<0.01$; testing cell source $F_{3,936}=1.1$, $p=0.36$; training cell number × testing cell source interaction $F_{15,936}=0.3$, $p=0.99$). When 20 cells were included in the machine learning training model, the Precision in predicting Ca^{2+} transients (**D**, $F_{3,156}=2.7$, $p=0.046$) and no Ca^{2+} transients (**E**, $F_{3,156}=0.2$, $p=0.93$) in testing cells from different session stage on different day reduced.

F-I, The training cell number, but not the testing cell source or their interactions, significantly affected the Recall in predicting Ca^{2+} transients (**F**, training cell number $F_{5,936}=126.0$, $p<0.01$; testing cell source $F_{3,936}=0.8$, $p=0.49$; training cell number × testing cell source interaction $F_{15,936}=0.3$, $p=0.99$). The number of cells,

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

the testing cell source, but not their interactions, significantly affected the Recall in predicting no Ca^{2+} transients (**G**, training cell number $F_{5,936}$=2.7, p=0.02; testing cell source $F_{3,936}$=11.4, p<0.01; training cell number × testing cell source interaction $F_{15,936}$=1.5, p=0.10). When 20 cells were included in the machine learning training model, the Recall in predicting Ca^{2+} transients (**H**, $F_{3,156}$=0.4, p=0.78) and no Ca^{2+} transients (**I**, $F_{3,156}$=3.3, p=0.02) in testing cells from different session stage on different day reduced.

J-M, The training cell number and the testing cell source, but not their interactions, significantly affected the F1 scores in predicting Ca^{2+} transients (**J**, training cell number $F_{5,936}$=130.0, p<0.01; testing cell source $F_{3,936}$=3.0, p=0.03; training cell number × testing cell source interaction $F_{15,936}$=0.7, p=0.74). Similarly, the number of cells and the testing cell source, but not their interactions, significantly affected the F1 scores in predicting no Ca^{2+} transients (**K**, training cell number $F_{5,936}$=32.3, p<0.01; testing cell source $F_{3,936}$=8.8, p<0.01; training cell number × testing cell source interaction $F_{15,936}$=1.2, p=0.28). When 20 cells were included in the machine learning training model, the Recall in predicting Ca^{2+} transients (**L**, $F_{3,156}$=2.7, p=0.048) and no Ca^{2+} transients (**M**, $F_{3,156}$=2.8, p=0.040) in testing cells from different session stage on different day reduced.

N, **O**, The training cell number, and the testing cell source, but not their interactions, significantly affected the macro F1 scores in predicting Ca^{2+} transients and no Ca^{2+} transients (**N**, training cell number $F_{5,936}$=128.5, p<0.01; testing cell source $F_{3,936}$=3.1, p=0.03; training cell number × testing cell source interaction

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

$F_{15,936}$=0.8, p=0.73). When 20 cells were included in the machine learning training model, the macro F1 scores in predicting Ca^{2+} transients and no Ca^{2+} transients (**O**, $F_{3,156}$=2.8, p=0.041) in testing cells from different session stage on different day reduced.

P, Legends showing the color-coded abbreviations for 4 experimental groups.

Data were analyzed by one-way ANOVA (**B**, **C**, **F**, **G**, **J**, **K**, **N**) or two-way ANOVA (**D**, **E**, **H**, **I**, **L**, **M**, **O**), followed by Bonferroni *post hoc* test. *, p<0.05; **, p<0.01. 40 testing cells in each group.

Figure 9.

A, Three strategies for sourcing testing cells (smSS::smDay, between mice, different regions, see more details in panel **P**)

B-E, The training cell number, and the testing cell source, but not their interactions, significantly affected the Precision in predicting Ca^{2+} transients (**B**, training cell number $F_{5,942}$=24.4, p<0.01; testing cell source $F_{2,942}$=19.2, p<0.01; training cell number × testing cell source interaction $F_{10,942}$=1.1, p=0.37) and the no Ca^{2+} transients (**C**, training cell number $F_{5,942}$=74.1, p<0.01; testing cell source $F_{2,942}$=11.6, p<0.01; training cell number × testing cell source interaction $F_{10,942}$=1.0, p=0.43). When 20 cells were included in the machine learning training model, the Precision in predicting Ca^{2+} transients (**D**, $F_{2,157}$=4.7, p=0.01) and no Ca^{2+} transients (**E**, $F_{2,157}$=6.2, p<0.001) in testing cells from different regions in different mice reduced.

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

F-I, The training cell number, but not the testing cell source or their interactions, significantly affected the Recall in predicting Ca^{2+} transients (**F**, training cell number $F_{5,942}$=145.6, p<0.01; testing cell source $F_{2,942}$=3.7, p=0.02; training cell number × testing cell source interaction $F_{10,942}$=0.7, p=0.68). The training cell number and the testing cell source, but not their interactions, significantly affected the Recall in predicting no Ca^{2+} transients (**G**, training cell number $F_{5,942}$=8.7, p<0.01; testing cell source $F_{2,942}$=22.9, p<0.01; training cell number × testing cell source interaction $F_{10,942}$=0.9, p=0.58). When 20 cells were included in the machine learning training model, the Recall in predicting Ca^{2+} transients (**H**, $F_{2,157}$=4.4, p=0.01) and no Ca^{2+} transients (**I**, $F_{2,157}$=5.3, p<0.01) in testing cells from different regions in different mice reduced.

J-M, The training cell number, and the testing cell source, but not their interactions, significantly affected the F1 scores in predicting Ca^{2+} transients (**J**, training cell number $F_{5,942}$=165.7, p<0.01; testing cell source $F_{2,942}$=17.6, p<0.01; training cell number × testing cell source interaction $F_{10,942}$=0.3, p=0.98) and the no Ca^{2+} transients (**K**, training cell number $F_{5,942}$=55.8, p<0.01; testing cell source $F_{2,942}$=40.4, p<0.01; training cell number × testing cell source interaction $F_{10,942}$=1.3, p=0.23). When 20 cells were included in the machine learning training model, the F1 scores in predicting Ca^{2+} transients (**L**, $F_{2,157}$=8.9, p<0.01) and no Ca^{2+} transients (**M**, $F_{2,157}$=9.3, p<0.01) in testing cells from different regions in different mice reduced.

N,**O**, The training cell number, and the testing cell source, but not their interactions, significantly affected the macro F1 scores in predicting Ca^{2+} transients and no

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Ca^{2+} transients (**N**, training cell number $F_{5,942}$=164.6, p<0.01; testing cell source $F_{2,942}$=18.3, p<0.01; training cell number × testing cell source interaction $F_{10,942}$=0.3, p=0.98). When 20 cells were included in the machine learning training model, the macro F1 scores in predicting Ca^{2+} transients and no Ca^{2+} transients (**O**, $F_{2,157}$=9.0, p<0.01) in testing cells from different regions in different mice reduced.

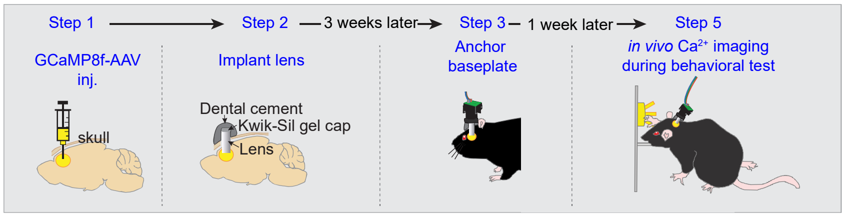
P, Legends showing the color-coded abbreviations for 4 experimental groups.

Data were analyzed by one-way ANOVA (**B**, **C**, **F**, **G**, **J**, **K**, **N**) or two-way ANOVA (**D**, **E**, **H**, **I**, **L**, **M**, **O**), followed by Bonferroni *post hoc* test. *, p<0.05; **, p<0.01. 40 testing cells in each group.

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Figure 1

Ca^{2+} imaging recordings on a freely moving mouse

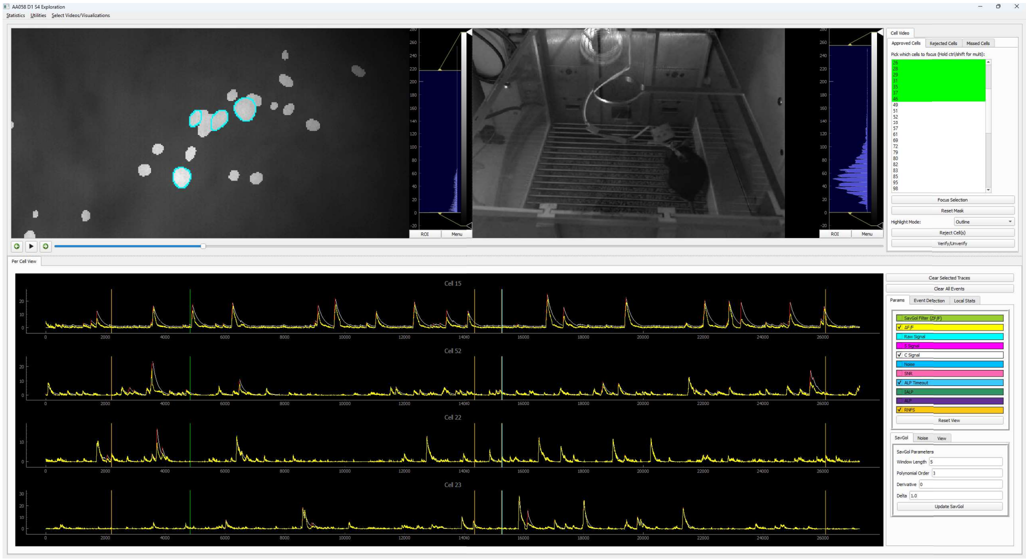


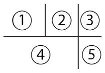
CalV2N: pipelines, such as miniAn (Zhou et al., 2022), CalmAn (Giovannucci et al., 2019), MIN1PIPE(Lu et al., 2018), developed to convert **Cal**cium **V**isual signals (i.e., Ca^{2+} imaging video files) **to N**umerical information.

CalTrig: **Cal**cium **Tr**ansient **I**dentifier **G**UI, facilitating data visualization, evaluating the performance of CalV2N, and identifying calcium transients.

Final statistics and results report

Figure 2.





① Ca2+ image video

② Behavioral video tracking

③ Cell list

④ Ca2+ trace

⑤ Trace toolbox

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Figure 3

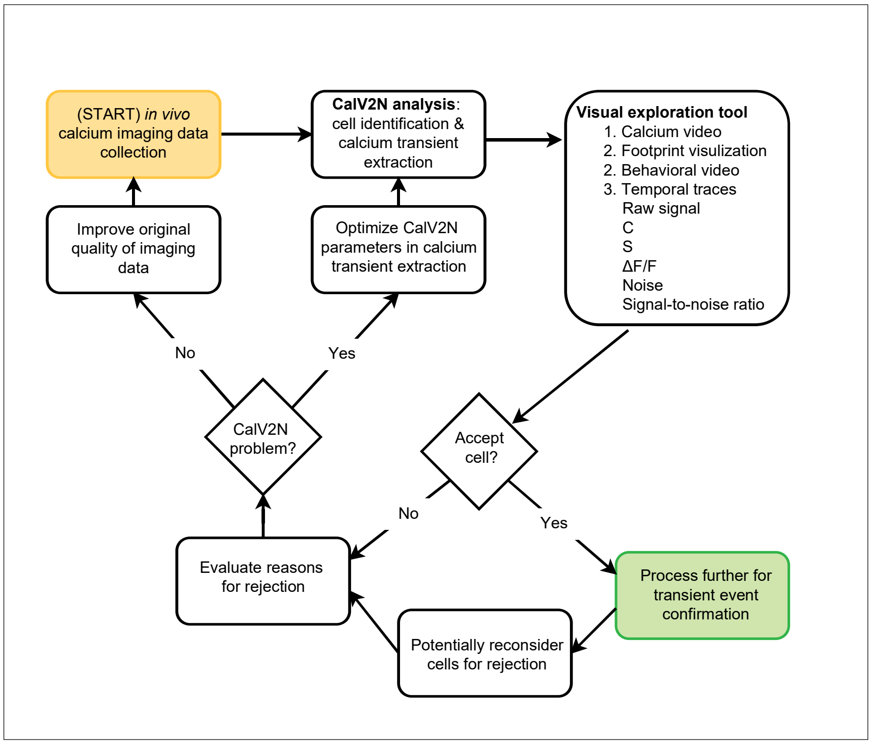
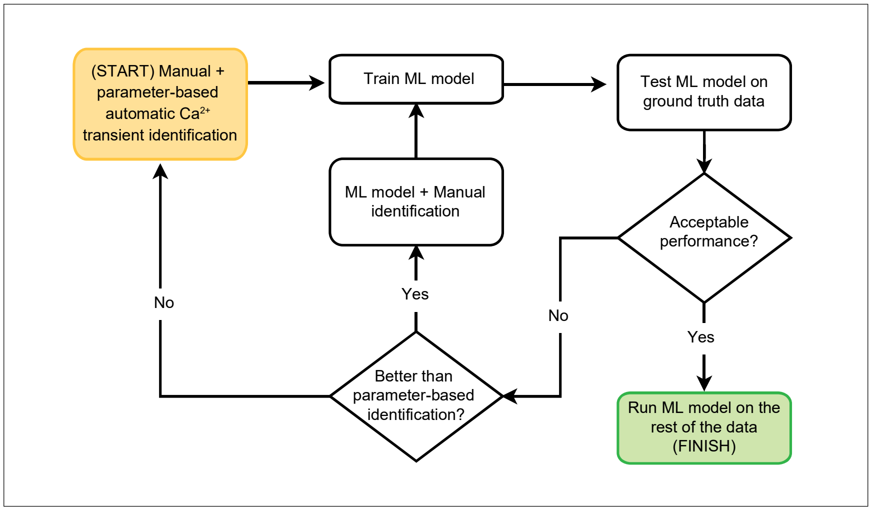
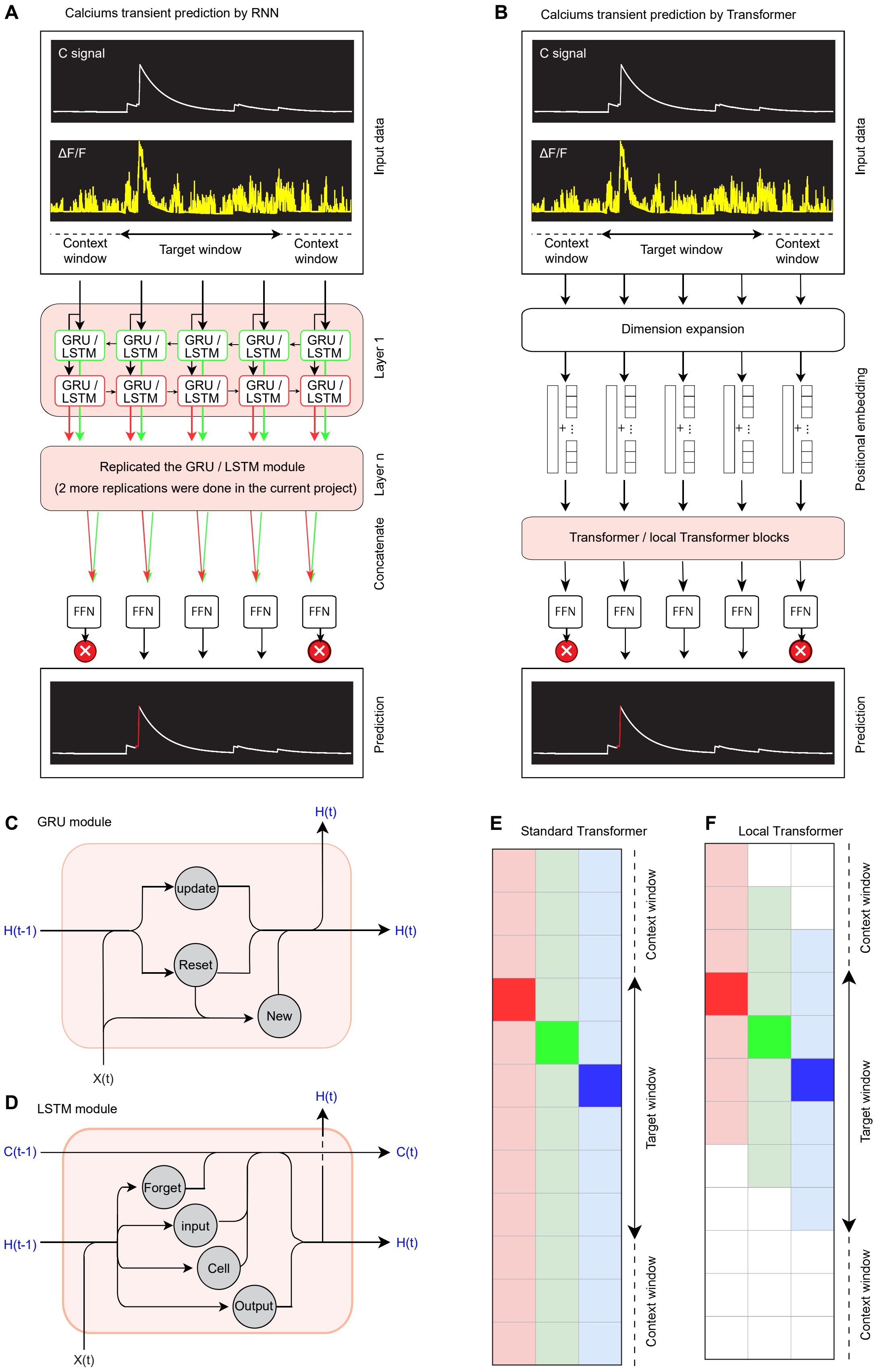

(START) *in vivo* calcium imaging data collection
CalV2N analysis: cell identification & calcium transient extraction
Visual exploration tool
1. Calcium video
2. Footprint visulization
2. Behavioral video
3. Temporal traces
Raw signal
C
S
ΔF/F
Noise
Signal-to-noise ratio
Improve original quality of imaging data
Optimize CalV2N parameters in calcium transient extraction
No
Yes
CalV2N problem?
Accept cell?
No
Yes
Evaluate reasons for rejection
Process further for transient event confirmation
Potentially reconsider cells for rejection

Figure 4


(START) Manual + parameter-based automatic Ca2+ transient identification
Train ML model
Test ML model on ground truth data
ML model + Manual identification
Acceptable performance?
No
Yes
No
Yes
Better than parameter-based identification?
Run ML model on the rest of the data (FINISH)

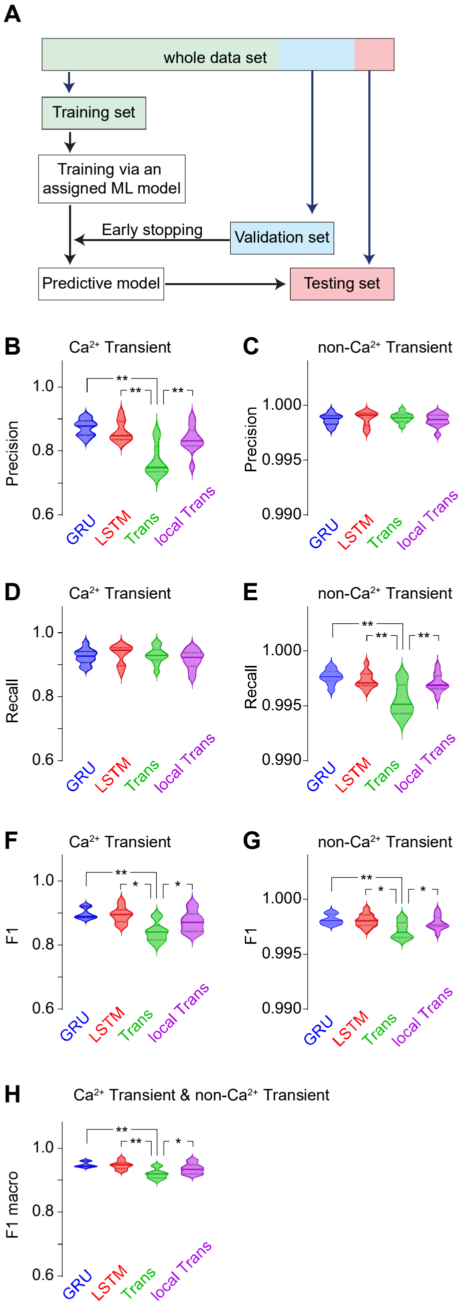
bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Figure 5



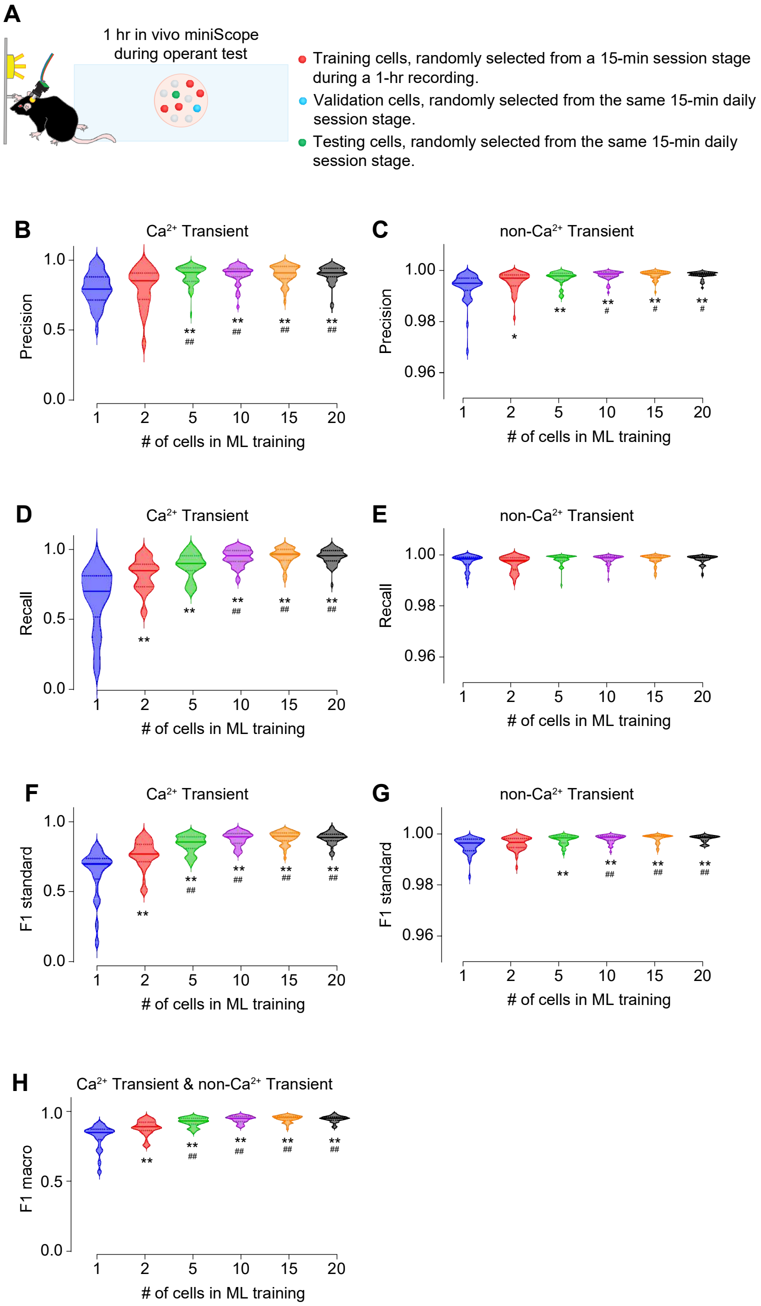
bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Figure 6



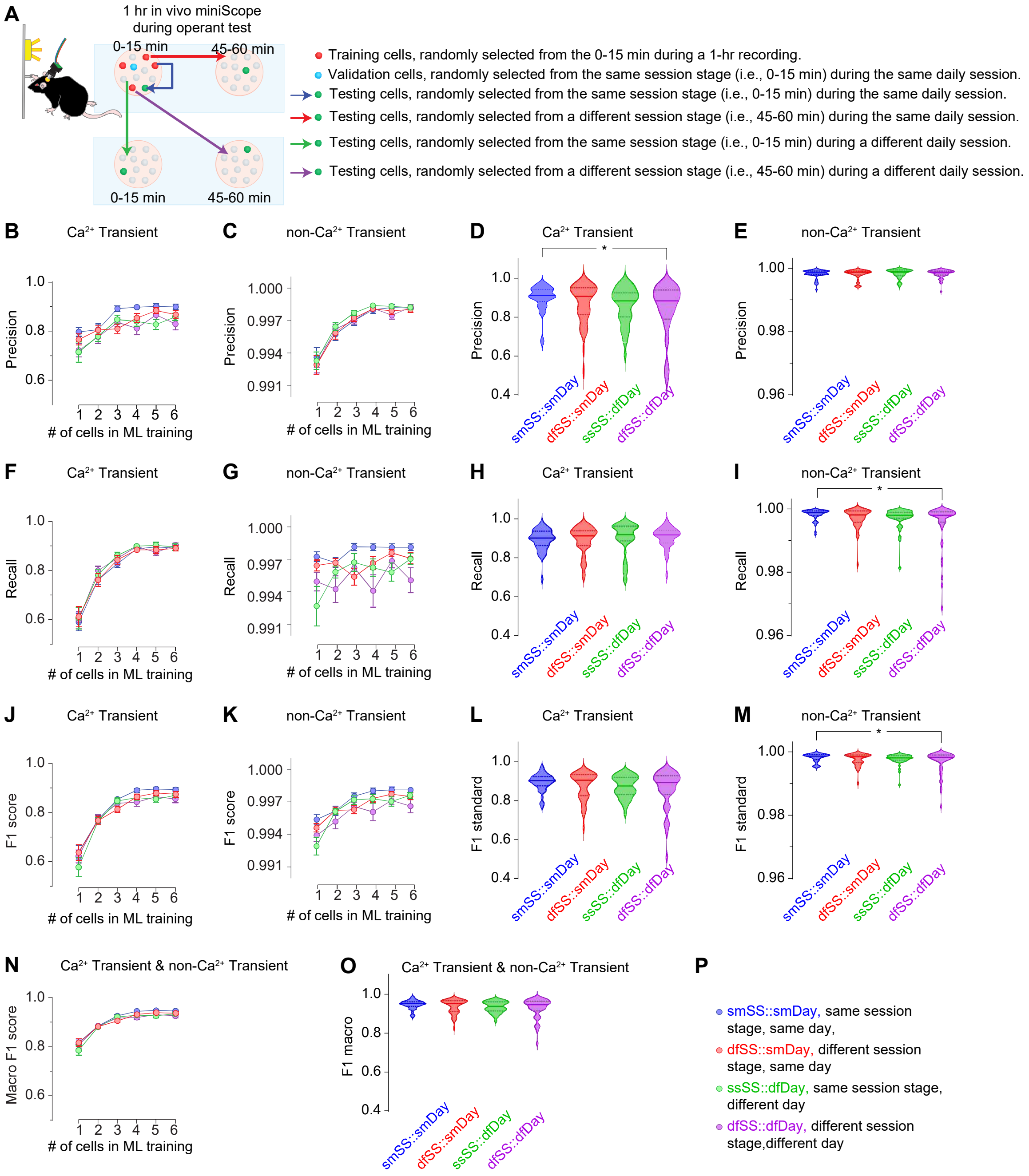
bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Figure 7



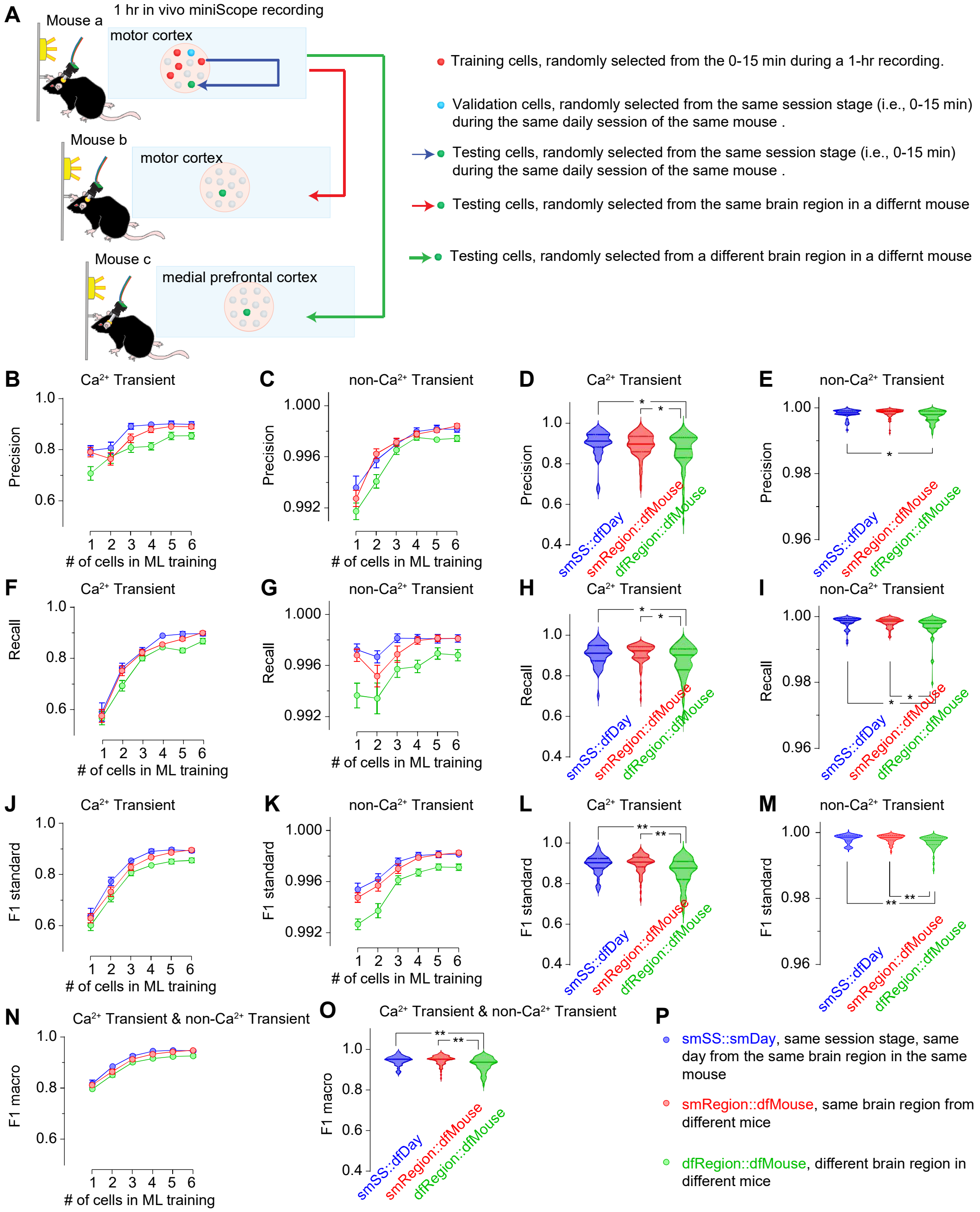
bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Figure 8



bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Figure 9



bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Figure S1. Data loading

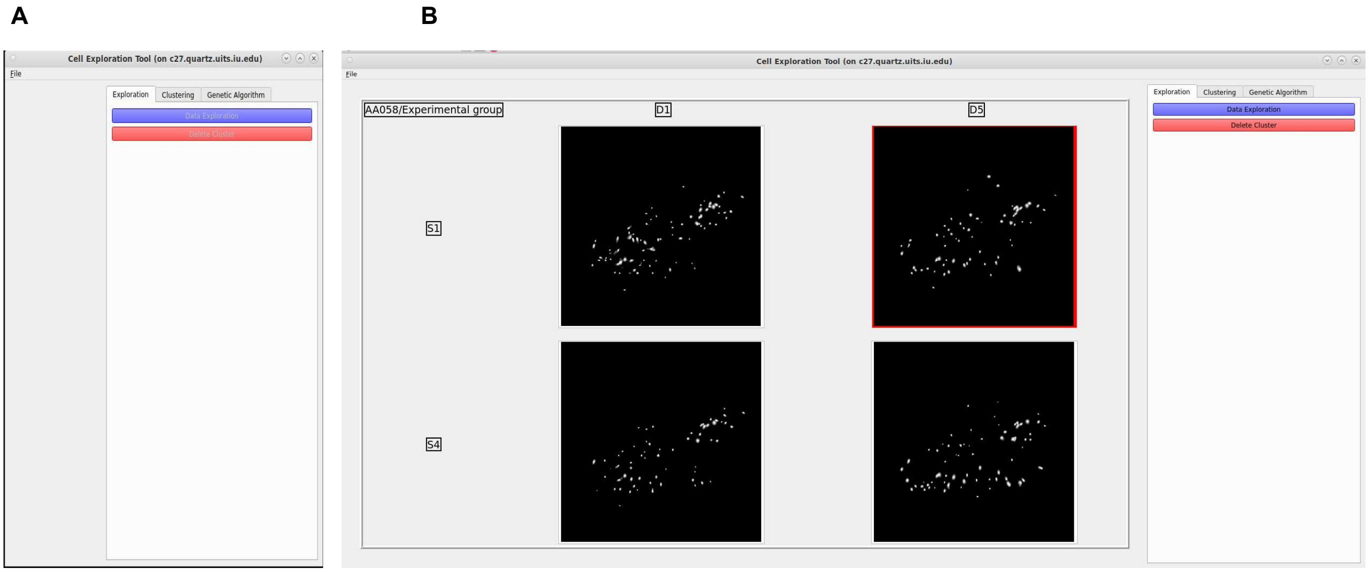
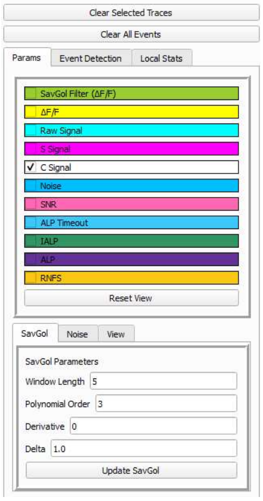


Figure S2. Parameter list



SavGol Parameters

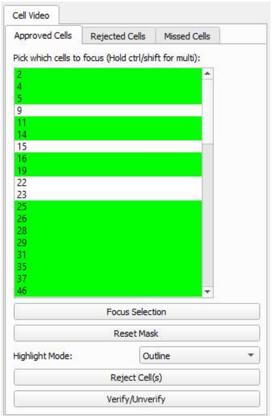
Window Length 5

Polynomial Order 3

Derivative 0

Delta 1.0

Figure S3. Cell verification



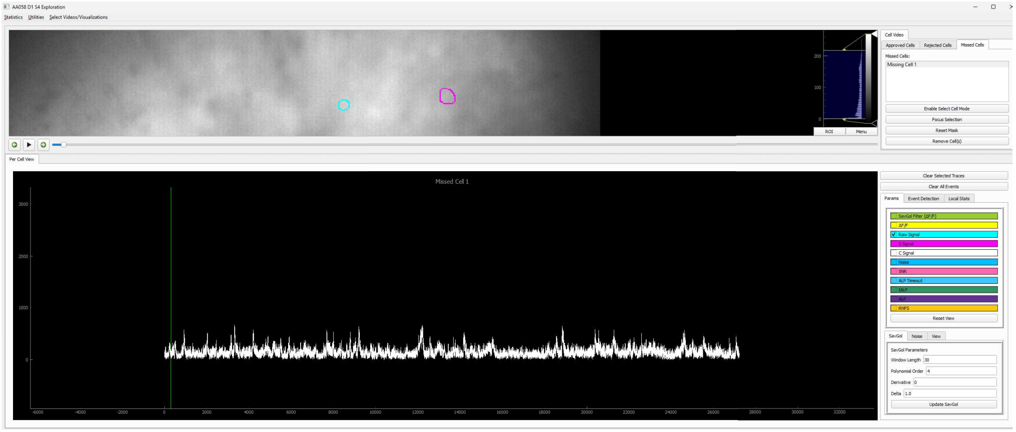
Pick which cells to focus (Hold ctrl/shift for multi):

2
4
5
9
11
14
15
16
19
22
23
25
26
28
29
31
35
37
46

Highlight Mode: Outline

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Figure S4. Add missing cell



Missed Cells:
Missing Cell 1

☐ SavGol Filter (ΔF/F)
☐ ΔF/F
☑ Raw Signal
☐ S Signal
☐ C Signal
☐ Noise
☐ SNR
☐ ALP Timeout
☐ IALP
☐ ALP
☐ RNFS

SavGol Parameters
Window Length 30
Polynomial Order 4
Derivative 0
Delta 1.0

Figure S5. Manual identification of Ca2+ transients

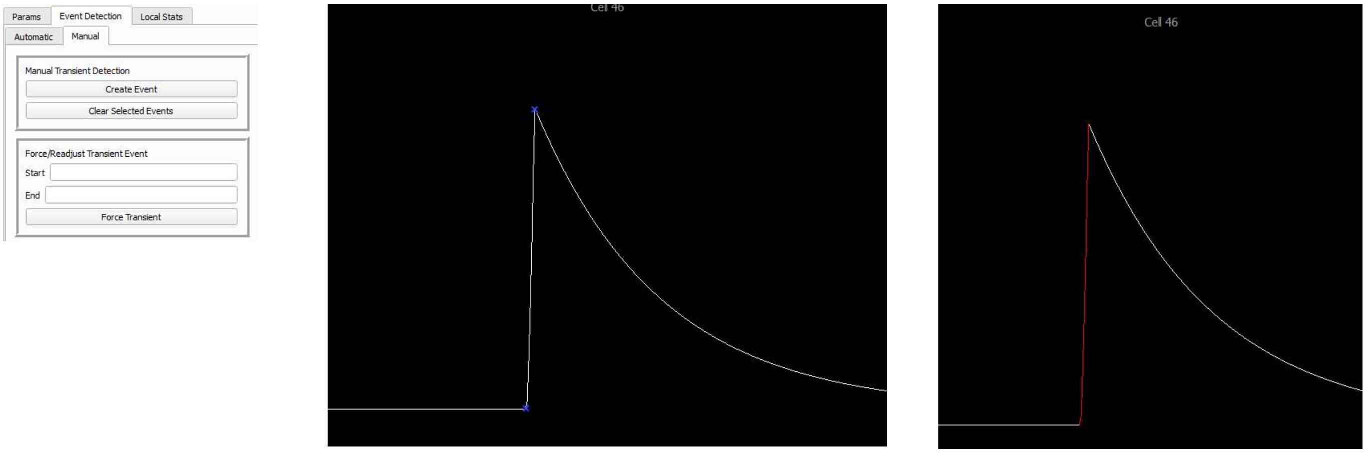
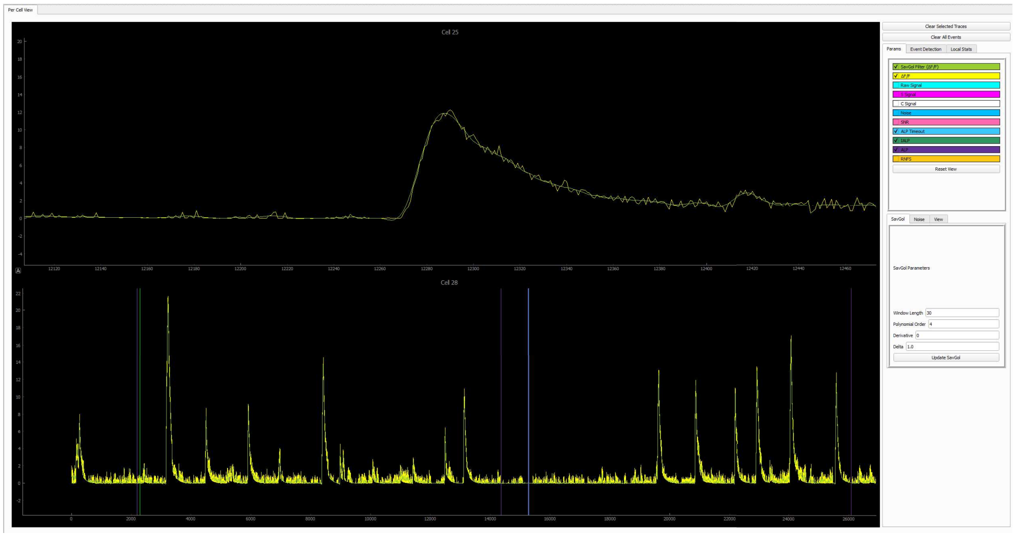


Figure S6. Savitzky–Golay filter

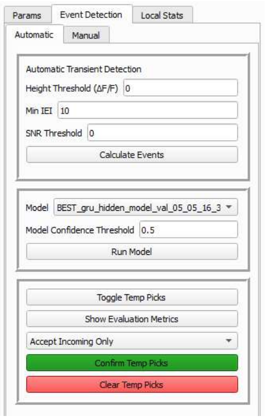


☑ SavGol Filter (ΔF/F)
☑ ΔF/F
☐ Raw Signal
☐ S Signal
☐ C Signal
☐ Noise
☐ SNR
☑ ALP Timeout
☑ IALP
☑ ALP
☐ RNFS

SavGol Parameters
Window Length 30
Polynomial Order 4
Derivative 0
Delta 1.0

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Figure S7. Automatic detection



Automatic Transient Detection

Height Threshold (ΔF/F) 0

Min IEI 10

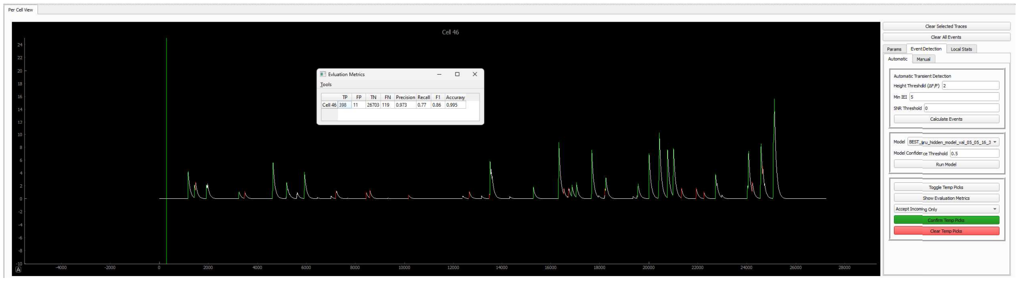
SNR Threshold 0

Model BEST_gru_hidden_model_val_05_05_16_3

Model Confidence Threshold 0.5

Accept Incoming Only

Figure S8. Machine learning evaluation



Evluation Metrics

Tools

	TP	FP	TN	FN	Precision	Recall	F1	Accuracy
Cell 46	398	11	26703	119	0.973	0.77	0.86	0.995

Automatic Transient Detection

Height Threshold (ΔF/F) 2

Min IEI 5

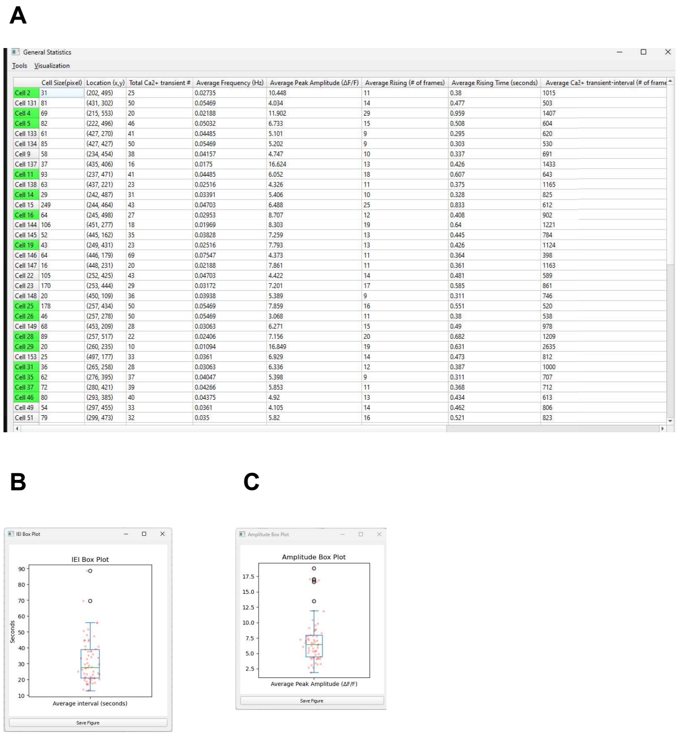
SNR Threshold 0

Model BEST_gru_hidden_model_val_05_05_16_3

Model Confidence Threshold 0.5

Accept Incoming Only

Figure S9. Animal-wide data export



General Statistics

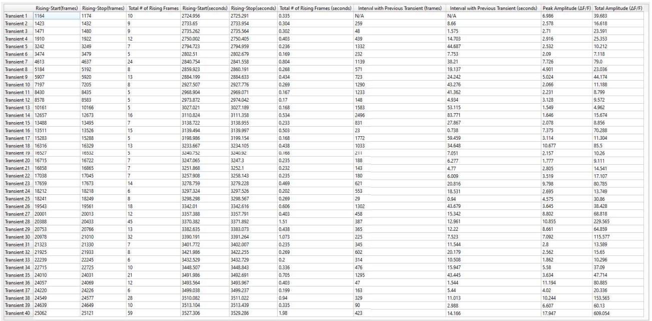
Tools Visualization

	Cell Size(pixel)	Location (x,y)	Total Ca2+ transient #	Average Frequency (Hz)	Average Peak Amplitude (ΔF/F)	Average Rising (# of frames)	Average Rising Time (seconds)	Average Ca2+ transient-interval (# of frame
Cell 2	31	(202, 495)	25	0.02735	10.448	11	0.38	1015
Cell 131	81	(431, 302)	50	0.05469	4.034	14	0.477	503
Cell 4	69	(215, 553)	20	0.02188	11.902	29	0.959	1407
Cell 5	82	(222, 496)	46	0.05032	6.733	15	0.508	604
Cell 133	61	(427, 270)	41	0.04485	5.101	9	0.295	620
Cell 134	85	(427, 427)	50	0.05469	5.202	9	0.303	530
Cell 9	58	(234, 454)	38	0.04157	4.747	10	0.337	691
Cell 137	37	(435, 406)	16	0.0175	16.624	13	0.426	1433
Cell 11	93	(237, 471)	41	0.04485	6.052	18	0.607	643
Cell 138	63	(437, 221)	23	0.02516	4.326	11	0.375	1165
Cell 14	29	(242, 487)	31	0.03391	5.406	10	0.328	825
Cell 15	249	(244, 464)	43	0.04703	6.488	25	0.833	612
Cell 16	64	(245, 498)	27	0.02953	8.707	12	0.408	902
Cell 144	106	(451, 277)	18	0.01969	8.303	19	0.64	1221
Cell 145	52	(445, 162)	35	0.03828	7.259	13	0.445	784
Cell 19	43	(249, 431)	23	0.02516	7.793	13	0.426	1124
Cell 146	64	(446, 179)	69	0.07547	4.373	11	0.364	398
Cell 147	16	(448, 231)	20	0.02188	7.861	11	0.361	1163
Cell 22	105	(252, 425)	43	0.04703	4.422	14	0.481	589
Cell 23	170	(253, 444)	29	0.03172	7.201	17	0.585	861
Cell 148	20	(450, 109)	36	0.03938	5.389	9	0.311	746
Cell 25	178	(257, 434)	50	0.05469	7.859	16	0.551	520
Cell 26	46	(257, 278)	50	0.05469	3.068	11	0.38	538
Cell 149	68	(453, 209)	28	0.03063	6.271	15	0.49	978
Cell 28	89	(257, 517)	22	0.02406	7.156	20	0.682	1209
Cell 29	20	(260, 235)	10	0.01094	16.849	19	0.631	2635
Cell 153	25	(497, 177)	33	0.0361	6.929	14	0.473	812
Cell 31	36	(265, 258)	28	0.03063	6.336	12	0.387	1000
Cell 35	62	(276, 395)	37	0.04047	5.398	9	0.311	707
Cell 37	72	(280, 421)	39	0.04266	5.853	11	0.368	712
Cell 46	80	(293, 385)	40	0.04375	4.92	13	0.434	613
Cell 49	54	(297, 455)	33	0.0361	4.105	14	0.462	806
Cell 51	79	(299, 473)	32	0.035	5.82	16	0.521	823

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

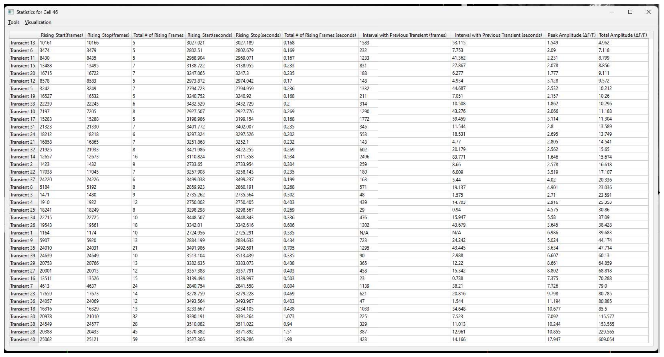
Figure S10. Cell-wide data export

A



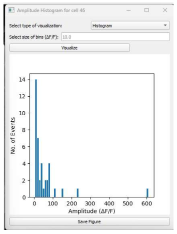
	Rising-Start(frames)	Rising-Stop(frames)	Total # of Rising Frames	Rising-Start(seconds)	Rising-Stop(seconds)	Total # of Rising Frames (seconds)	Interval with Previous Transient (frames)	Interval with Previous Transient (seconds)	Peak Amplitude (ΔF/F)	Total Amplitude (ΔF/F)
Transient 1	1164	1174	10	2724.956	2725.291	0.335	N/A	N/A	6.986	39.683
Transient 2	1423	1432	9	2733.65	2733.954	0.304	259	8.66	2.578	16.618
Transient 3	1471	1480	9	2735.262	2735.564	0.302	48	1.575	2.71	23.591
Transient 4	1910	1922	12	2750.002	2750.405	0.403	439	14.703	2.916	25.353
Transient 5	3242	3249	7	2794.723	2794.959	0.236	1332	44.687	2.532	10.212
Transient 6	3474	3479	5	2802.51	2802.679	0.169	232	7.753	2.09	7.118
Transient 7	4613	4637	24	2840.754	2841.558	0.804	1139	38.21	7.726	79.0
Transient 8	5184	5192	8	2859.923	2860.191	0.268	571	19.137	4.901	23.036
Transient 9	5907	5920	13	2884.199	2884.633	0.434	723	24.242	5.024	44.174
Transient 10	7197	7205	8	2927.507	2927.776	0.269	1290	43.276	2.066	11.188
Transient 11	8430	8435	5	2968.904	2969.071	0.167	1233	41.362	2.231	8.799
Transient 12	8578	8583	5	2973.872	2974.042	0.17	148	4.934	3.128	9.572
Transient 13	10161	10166	5	3027.021	3027.189	0.168	1583	53.115	1.549	4.962
Transient 14	12657	12673	16	3110.824	3111.358	0.534	2496	83.771	1.646	15.674
Transient 15	13488	13495	7	3138.722	3138.955	0.233	831	27.867	2.078	8.856
Transient 16	13511	13526	15	3139.494	3139.997	0.503	23	0.738	7.375	70.288
Transient 17	15283	15288	5	3198.986	3199.154	0.168	1772	59.459	3.114	11.304
Transient 18	16316	16329	13	3233.667	3234.105	0.438	1033	34.648	10.677	85.5
Transient 19	16527	16532	5	3240.752	3240.92	0.168	211	7.051	2.157	10.26
Transient 20	16715	16722	7	3247.065	3247.3	0.235	188	6.277	1.777	9.111
Transient 21	16858	16865	7	3251.868	3252.1	0.232	143	4.77	2.805	14.541
Transient 22	17038	17045	7	3257.908	3258.143	0.235	180	6.009	3.519	17.107
Transient 23	17659	17673	14	3278.759	3279.228	0.469	621	20.816	9.798	80.785
Transient 24	18212	18218	6	3297.324	3297.526	0.202	553	18.531	2.695	13.749
Transient 25	18241	18249	8	3298.298	3298.567	0.269	29	0.94	4.575	30.86
Transient 26	19543	19561	18	3342.01	3342.616	0.606	1302	43.679	3.645	38.428
Transient 27	20001	20013	12	3357.388	3357.791	0.403	458	15.342	8.802	68.818
Transient 28	20388	20433	45	3370.382	3371.892	1.51	387	12.961	10.855	229.565
Transient 29	20753	20766	13	3382.635	3383.073	0.438	365	12.22	8.661	64.859
Transient 30	20978	21010	32	3390.191	3391.264	1.073	225	7.523	7.092	115.577
Transient 31	21323	21330	7	3401.772	3402.007	0.235	345	11.544	2.8	13.589
Transient 32	21925	21933	8	3421.986	3422.255	0.269	602	20.179	2.562	15.65
Transient 33	22239	22245	6	3432.529	3432.729	0.2	314	10.508	1.862	10.296
Transient 34	22715	22725	10	3448.507	3448.843	0.336	476	15.947	5.58	37.09
Transient 35	24010	24031	21	3491.986	3492.691	0.705	1295	43.445	3.634	47.714
Transient 36	24057	24069	12	3493.564	3493.967	0.403	47	1.544	11.194	80.885
Transient 37	24220	24226	6	3499.038	3499.237	0.199	163	5.44	4.02	20.336
Transient 38	24549	24577	28	3510.082	3511.022	0.94	329	11.013	10.244	153.565
Transient 39	24639	24649	10	3513.104	3513.439	0.335	90	2.988	6.607	60.13
Transient 40	25062	25121	59	3527.306	3529.286	1.98	423	14.166	17.947	609.054

B



	Rising-Start(frames)	Rising-Stop(frames)	Total # of Rising Frames	Rising-Start(seconds)	Rising-Stop(seconds)	Total # of Rising Frames (seconds)	Interval with Previous Transient (frames)	Interval with Previous Transient (seconds)	Peak Amplitude (ΔF/F)	Total Amplitude (ΔF/F)
Transient 13	10161	10166	5	3027.021	3027.189	0.168	1583	53.115	1.549	4.962
Transient 6	3474	3479	5	2802.51	2802.679	0.169	232	7.753	2.09	7.118
Transient 11	8430	8435	5	2968.904	2969.071	0.167	1233	41.362	2.231	8.799
Transient 15	13488	13495	7	3138.722	3138.955	0.233	831	27.867	2.078	8.856
Transient 20	16715	16722	7	3247.065	3247.3	0.235	188	6.277	1.777	9.111
Transient 12	8578	8583	5	2973.872	2974.042	0.17	148	4.934	3.128	9.572
Transient 5	3242	3249	7	2794.723	2794.959	0.236	1332	44.687	2.532	10.212
Transient 19	16527	16532	5	3240.752	3240.92	0.168	211	7.051	2.157	10.26
Transient 33	22239	22245	6	3432.529	3432.729	0.2	314	10.508	1.862	10.296
Transient 10	7197	7205	8	2927.507	2927.776	0.269	1290	43.276	2.066	11.188
Transient 17	15283	15288	5	3198.986	3199.154	0.168	1772	59.459	3.114	11.304
Transient 31	21323	21330	7	3401.772	3402.007	0.235	345	11.544	2.8	13.589
Transient 24	18212	18218	6	3297.324	3297.526	0.202	553	18.531	2.695	13.749
Transient 21	16858	16865	7	3251.868	3252.1	0.232	143	4.77	2.805	14.541
Transient 32	21925	21933	8	3421.986	3422.255	0.269	602	20.179	2.562	15.65
Transient 14	12657	12673	16	3110.824	3111.358	0.534	2496	83.771	1.646	15.674
Transient 2	1423	1432	9	2733.65	2733.954	0.304	259	8.66	2.578	16.618
Transient 22	17038	17045	7	3257.908	3258.143	0.235	180	6.009	3.519	17.107
Transient 37	24220	24226	6	3499.038	3499.237	0.199	163	5.44	4.02	20.336
Transient 8	5184	5192	8	2859.923	2860.191	0.268	571	19.137	4.901	23.036
Transient 3	1471	1480	9	2735.262	2735.564	0.302	48	1.575	2.71	23.591
Transient 4	1910	1922	12	2750.002	2750.405	0.403	439	14.703	2.916	25.353
Transient 25	18241	18249	8	3298.298	3298.567	0.269	29	0.94	4.575	30.86
Transient 34	22715	22725	10	3448.507	3448.843	0.336	476	15.947	5.58	37.09
Transient 26	19543	19561	18	3342.01	3342.616	0.606	1302	43.679	3.645	38.428
Transient 1	1164	1174	10	2724.956	2725.291	0.335	N/A	N/A	6.986	39.683
Transient 9	5907	5920	13	2884.199	2884.633	0.434	723	24.242	5.024	44.174
Transient 35	24010	24031	21	3491.986	3492.691	0.705	1295	43.445	3.634	47.714
Transient 39	24639	24649	10	3513.104	3513.439	0.335	90	2.988	6.607	60.13
Transient 29	20753	20766	13	3382.635	3383.073	0.438	365	12.22	8.661	64.859
Transient 27	20001	20013	12	3357.388	3357.791	0.403	458	15.342	8.802	68.818
Transient 16	13511	13526	15	3139.494	3139.997	0.503	23	0.738	7.375	70.288
Transient 7	4613	4637	24	2840.754	2841.558	0.804	1139	38.21	7.726	79.0
Transient 23	17659	17673	14	3278.759	3279.228	0.469	621	20.816	9.798	80.785
Transient 36	24057	24069	12	3493.564	3493.967	0.403	47	1.544	11.194	80.885
Transient 18	16316	16329	13	3233.667	3234.105	0.438	1033	34.648	10.677	85.5
Transient 30	20978	21010	32	3390.191	3391.264	1.073	225	7.523	7.092	115.577
Transient 38	24549	24577	28	3510.082	3511.022	0.94	329	11.013	10.244	153.565
Transient 28	20388	20433	45	3370.382	3371.892	1.51	387	12.961	10.855	229.565
Transient 40	25062	25121	59	3527.306	3529.286	1.98	423	14.166	17.947	609.054

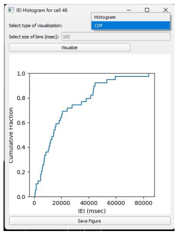
C



Select type of visualization: Histogram

Select size of bins (ΔF/F): 10.0

D



Select type of visualization: CDF

Select size of bins (msec): 100

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

Supplementary Information

CalTrig: A GUI-Based Machine Learning Approach for Decoding Neuronal Calcium Transients in Freely Moving Rodents

Michal A. Lange[1], Yingying Chen[1], Haoying Fu[1], Amith Korada[1], Changyong Guo[1], Yao-Ying Ma[1,2*]

[1], Department of Pharmacology and Toxicology, Indiana University School of Medicine, Indianapolis, IN 46202, USA.

[2], Stark Neurosciences Research Institute, Indiana University School of Medicine, Indianapolis, IN 46202, USA.

***Corresponding author**: Dr. Yao-Ying Ma

Signal-to-Noise Estimation

While verifying transient events, occasional spikes were detected due to background fluctuations rather than specific cellular activity, and in rare cases, noise was misinterpreted as calcium signals. Contemporary methods often use z-scores to estimate noise, but this global noise estimation lacks robustness when noise fluctuates over time, potentially leading to the rejection of valid transients during low-noise periods.

To address this concern, we apply the Savitzky–Golay (SavGol) filter (**Fig. S6**), which smooths noisy data while preserving important features such as peaks and edges (Steinier, Termonia et al. 1972, Dai, Selesnick et al. 2017). This filter fits successive

bioRxiv preprint doi: https://doi.org/10.1101/2024.09.30.615860; this version posted October 1, 2024. The copyright holder for this preprint (which was not certified by peer review) is the author/funder, who has granted bioRxiv a license to display the preprint in perpetuity. It is made available under aCC-BY-NC-ND 4.0 International license.

polynomial functions to adjacent data points, making it ideal for time-series data where preserving peak shape is crucial. In our pilot studies, where we compared the original ΔF/F with the smoothed ΔF/F using CalTrig's data exploration tools, the Ca2+ transients were well-preserved. Therefore, the SavGol filter was chosen as the primary smoothing filter.

Noise estimation for each image frame is calculated by the difference between the original and SavGol-filtered ΔF/F:

$$Sig_{noise}(t) \approx \left| Sig_{\frac{\Delta F}{F}} - savgol\left(Sig_{\frac{\Delta F}{F}}\right) \right|$$

To further refine the noise, we apply a rolling window function, preventing small value overlaps between the smoothed and non-smoothed signals. The user can choose smoothing types (average, median, max) and adjust the rolling window size (20 frames is a recommended starting point). SNR is calculated by dividing the smoothed signal by the estimated noise, with a minimum cap (e.g., 0.1) to avoid ineffective or exaggerated values. The resulting SNR effectively mirrors the ΔF/F, proportionally accentuated or diminished by the noise.

The parameters for the SavGol filter and noise smoothing can be adjusted under the "SavGol" and "Noise" subtabs in the Trace toolbox window. Visualization options for the SavGol-filtered ΔF/F, noise, and SNR are available under the "Params" subtab for easy analysis in the Ca2+ trace window.