

Allen Brain Observatory: Visual Behavior 2P

Technical Whitepaper

SECTION A: OVERVIEW	3
VISUAL BEHAVIOR ALLEN BRAIN OBSERVATORY PIPELINE	3
CHANGE DETECTION TASK.....	3
2-PHOTON CALCIUM IMAGING	4
EXPERIMENTAL DESIGN.....	5
DATASET VARIANTS.....	5
DATA STRUCTURE AND TERMINOLOGY	7
SECTION B: MICE & SURGERY	8
MICE	8
SURGERY.....	8
SECTION C: INTRINSIC SIGNAL IMAGING	10
DATA ACQUISITION	10
APPARATUS.....	10
ISI QUALITY CONTROL.....	14
SECTION D: BEHAVIOR	15
WATER RESTRICTION AND HABITUATION	15
APPARATUS.....	15
CHANGE DETECTION TASK.....	16
BEHAVIOR METRICS.....	18
RUNNING SPEED CALCULATION	19
SECTION E: IN VIVO 2-PHOTON CALCIUM IMAGING	21
HARDWARE & INSTRUMENTATION	21
VISUAL STIMULATION DURING IN VIVO 2-PHOTON IMAGING.....	22
MOUSE BODY AND EYE MONITORING.....	22
DATA SYNCHRONIZATION	23
WORKFLOW SEQUENCING ENGINE (WSE)	24
DATA COLLECTION.....	24
QUALITY CONTROL FOR 2-PHOTON CALCIUM IMAGING	26

SECTION F: DATA PROCESSING..... 28

DEWARPING 28

MOTION CORRECTION 29

CELL SEGMENTATION..... 31

CROSSTALK REMOVAL IN MULTISCOPE DATA 32

ROI FILTERING 34

DEMIXING TRACES FROM OVERLAPPING ROIS..... 35

NEUROPIIL SUBTRACTION 36

DF/F CALCULATION 38

SESSION TO SESSION CELL MATCHING 38

EVENT DETECTION..... 42

EYE TRACKING DATA PROCESING 43

REFERENCES 45

SECTION A: OVERVIEW

To further our understanding of the neural basis of behavior, the Visual Behavior project utilized the Allen Brain Observatory platform for *in vivo* 2-photon calcium imaging to collect a large-scale, highly standardized dataset consisting of recordings of neural activity in mice performing a visually guided task.

A key aspect of the experimental design involved the repeated targeting of the same populations of neurons over multiple days, allowing analysis of single cell and population changes across varying behavioral and sensory conditions, including stimulus novelty and fluctuations in motivation. Accordingly, this dataset can be used to investigate how sensory stimuli and task performance are represented by patterns of activity in the visual cortex, and how this activity is influenced by experience and engagement.

This dataset includes neural and behavioral measurements from 82 mice, including 3021 behavior training sessions and 551 *in vivo* imaging sessions, resulting in longitudinal recordings from 34,619 cortical cells. These data are made openly accessible, with all recorded timeseries, behavioral events, and experimental metadata conveniently packaged in Neurodata Without Borders (NWB) files that can be accessed and analyzed using our open-source Python software package, the AllenSDK.

VISUAL BEHAVIOR ALLEN BRAIN OBSERVATORY PIPELINE



Figure 1. Visual Behavior Allen Brain Observatory. Each icon represents a distinct experimental step of the pipeline that is performed by a different team of technicians according to a series of standard operating procedures.

CHANGE DETECTION TASK

The Visual Behavior dataset is built upon a change detection behavioral task. Briefly, in this go/no-go task, mice are presented with a continuous series of briefly presented stimuli and they earn water rewards by correctly reporting when the identity of the image changes.

Mice undergo standardized training in the behavior facility prior to imaging, in which they first learn to detect orientation changes using static grating stimuli, then transition to flashed gratings, and subsequently learn to detect changes in the identity of natural scene images. Each session included 8 images, for a total of 64 possible transitions. The images used in this study were a selected subset of the natural scenes stimulus set used in the Visual Coding Allen Brain Observatory.

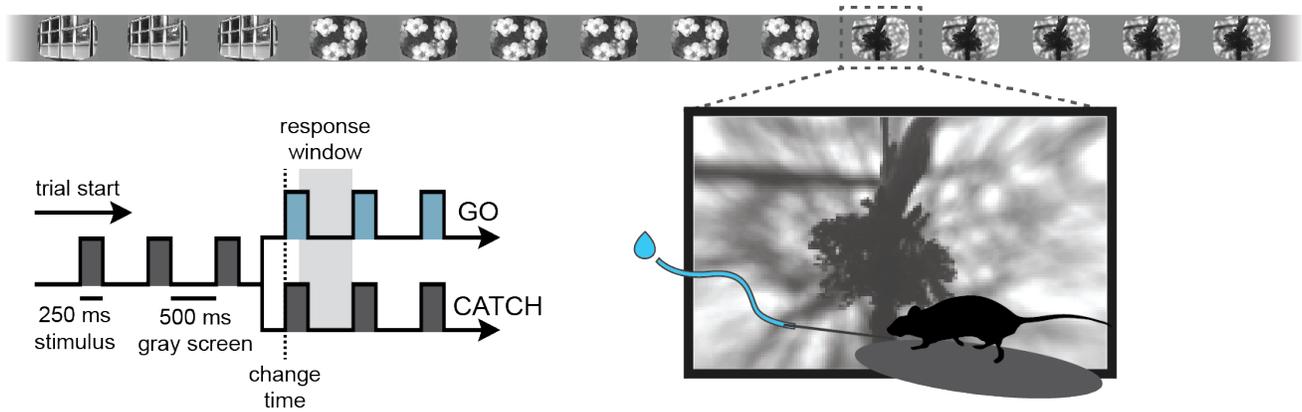


Figure 2. Change detection task. Top row shows the continuous stream of stimulus presentations (250ms per stimulus), with inter stimulus interval gray screen (500ms) that are displayed to the mouse during a behavior session. Lower left shows the two trial types in the task, “go” trials where the stimulus identity changes and the mouse must lick within a 750ms response window to earn a water reward, and “catch” trials where the image identity doesn’t change and false alarm licking behavior is quantified. Bottom right shows the behavioral setup, with stimuli displayed on a monitor facing the right eye of the mouse, a lick spout for response detection via a capacitive sensor and water reward delivery, and a running wheel.

2-PHOTON CALCIUM IMAGING

This dataset includes single- and multi-plane 2-photon calcium imaging recordings from the visual cortex of transgenic mice expressing the calcium indicator GCaMP6 in populations of excitatory and inhibitory neurons. Microscopes include Scientifica VivoScope systems for single-plane imaging, or a modified Mesoscope system (Multiscope) for multi-plane imaging.

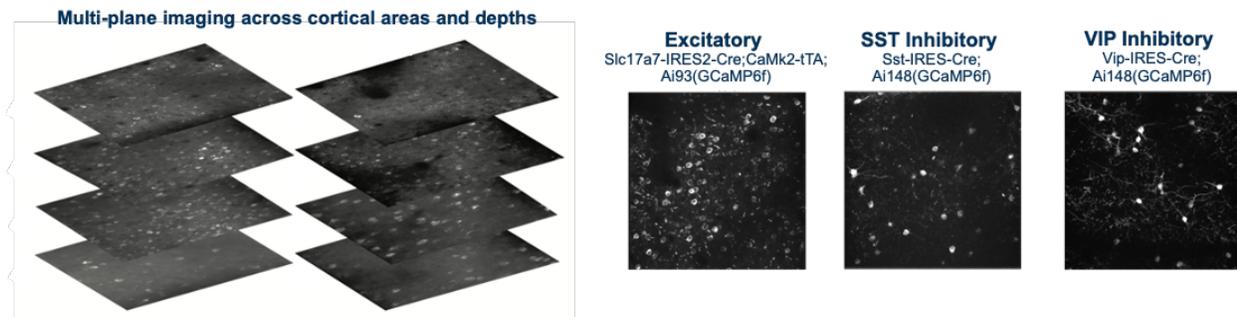


Figure 3. in vivo calcium imaging. Left, imaging configuration used for multi-plane imaging experiments, where imaging planes across 4 cortical depths in each of 2 areas were acquired in a given session. Right, maximum intensity projection images for the 3 major cell classes imaged, using a pan-excitatory driver line (Slc17a7-IRES2-Cre), a somatostatin expressing driver line (Sst-IRES-Cre) and a vasoactive intestinal peptide expressing driver line (Vip-IRES-Cre) crossed with a GCaMP6 reporter line (Ai93 or Ai148).

Once mice are well-trained in the task, they transition to performing the task while being imaged with a 2-photon microscope to enable simultaneous measurement of neural activity and behavior. During the imaging portion of the experiment, mice perform the task with the set of eight natural scene images they viewed during training, as well as a novel set of eight images that they had not seen before, in separate sessions. This allows evaluation of the impact of novelty on neural coding for stimulus and behavioral information. Mice also undergo passive viewing sessions during the imaging stage, where the mice are given their daily water before the session (and are thus satiated) and view the task stimuli with the lick spout retracted so they are unable to earn water rewards. Lastly, during imaging sessions only, stimuli were omitted with a 5% probability, disrupting the expected cadence of flashed image presentations. Stimulus changes and the stimulus immediately preceding the change were never omitted.

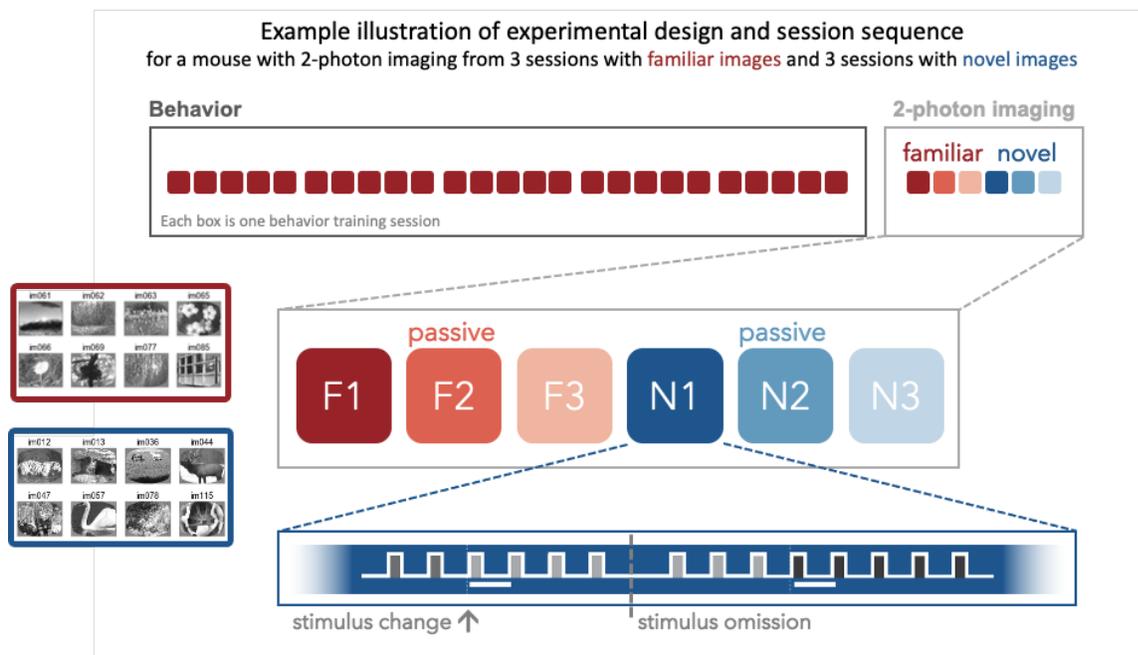


Figure 4. Experimental design. Each mouse was trained in the behavior facility until task performance reached criterion, then transferred to the 2-photon imaging portion of the experiment in which they underwent several different types of sessions. Neural activity was measured during task performance with the training image set (red), which was highly familiar to the mouse, as well as in response to a novel set of images (blue). Interleaved between active behavior sessions were passive sessions where the mouse was satiated and could not earn rewards. During each imaging session, the regular cadence of stimulus presentations was occasionally disrupted by a stimulus omission, which occurred randomly on 5% of non-change stimuli.

DATASET VARIANTS

To ensure that any observed effects of stimulus novelty are due to lack of experience with those images, rather than properties of the specific images that were used, we trained some mice with the opposite stimulus configuration such that the training set for these mice was the image set that was novel for a different group of mice (compare dataset variants VisualBehavior and VisualBehaviorTask1B in Figure 5). This control was performed using the single-plane 2-photon imaging system.

Visual Behavior 2P Datasets

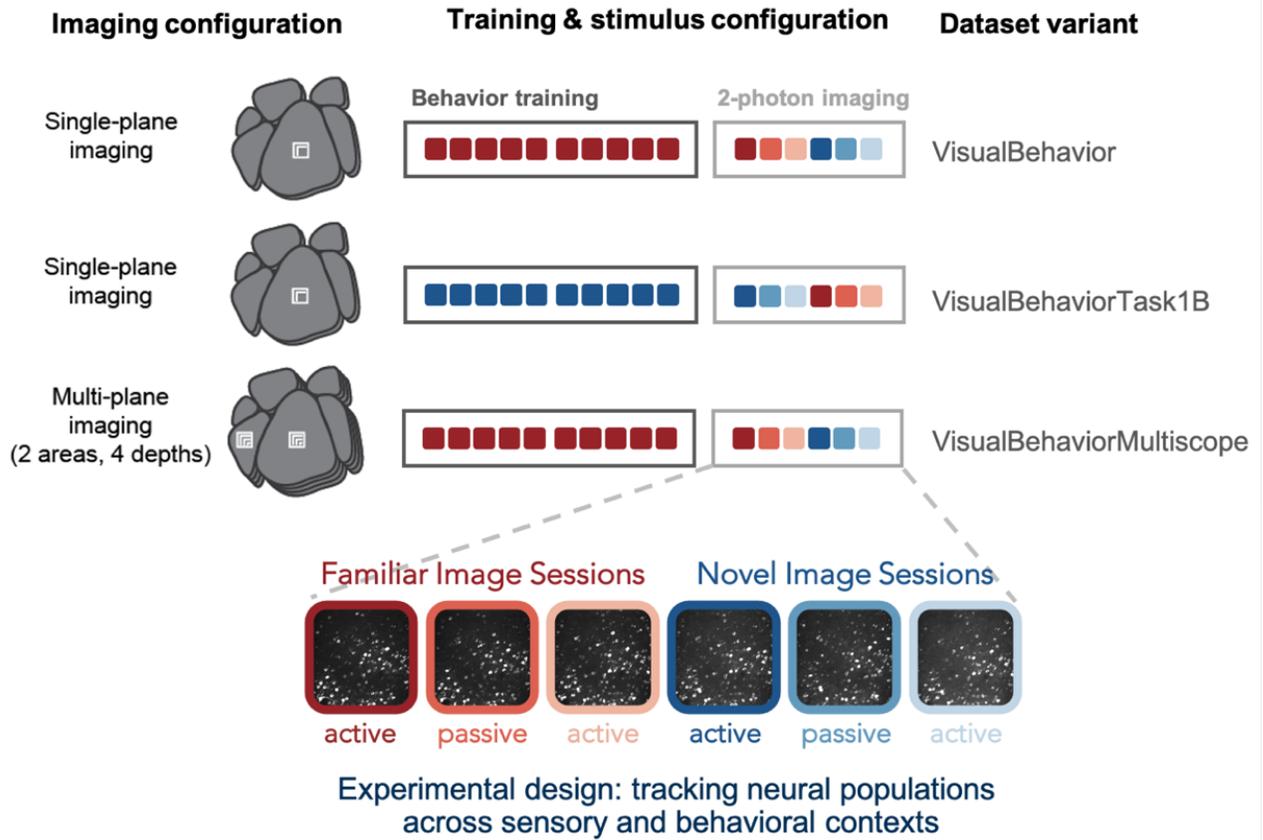


Figure 5. Dataset variants. Different groups of mice were subject to slight variations in experimental conditions. Single-plane imaging was performed on two groups of mice, each trained with a different image set, such that the novel image set for one group was the training set for the other group and vice versa, to control for the impact of stimulus features and novelty effects. A third group of mice was imaged using multi-plane imaging, where 4 depths were recorded from 2 cortical areas (VISp and VISl) in each session. Each of these variants has a unique name, also called a “project code” in the AllenSDK, to distinguish them from each other. Bottom row shows a key component of the experimental design – each population of neurons was imaged across multiple session types, to track changes in neural activity between familiar and novel image sessions, and between passive viewing of stimuli and active behavioral performance.

Dataset variant	Transgenic Line	# mice	# sessions	# imaging planes	# unique cells
VisualBehavior	Slc17a7;Ai93	17	103	103	5,915
	Slc17a7;Ai94	7	50	50	9,743
	Sst;Ai148	4	26	26	84
	Vip;Ai148	7	46	46	380
VisualBehaviorTask1B	Slc17a7;Ai93	14	95	95	4,672
	Slc17a7;Ai94	1	6	6	650
	Sst;Ai148	7	50	50	142
	Vip;Ai148	6	42	42	186
VisualBehaviorMultiscope	Slc17a7;Ai93	6	43	265	10,452
	Sst;Ai148	4	30	149	559
	Vip;Ai148	9	60	333	1,836
TOTAL		82	551	1,165	34,619

Table 1. Numbers of mice, sessions, imaging and uniquely recorded neurons for each dataset variant and transgenic line imaged.

DATA STRUCTURE AND TERMINOLOGY

Due to our experimental design, involving repeated imaging of the same population of cells over multiple days, the use of both single- and multi-plane imaging systems, and our strict quality control (QC) criteria that result in failures & retakes of specific session types, the complete 2-photon dataset for a given mouse can take many forms. The terminology described below is used to reference different subsets of the dataset, and these definitions apply to terms used in the AllenSDK, associated data manifest tables, and contents of NWB files.

The data collected in a single continuous recording is defined as a **session**. For single-plane imaging experiments, there is only one imaging plane (referred to as an **experiment**) per session. For multi-plane imaging experiments, there can be up to 8 imaging planes (8 experiments) per session. Due to our QC process, not all multi-plane imaging sessions have exactly 8 experiments, as some imaging planes did not meet our QC criteria.

We aimed to track the activity of single neurons across multiple **session types** (familiar and novel image sets, passive and active behavior sessions; described in the overview) by targeting the same population of neurons over multiple recording sessions, with only one session recorded per day for a given mouse. The collection of imaging sessions for a given population of cells, belonging to a single imaging plane measured across days, is called a **container**. A container can include between 3 and 11 separate sessions for a given imaging plane. Accordingly, the session types available for a given container (corresponding to a specific imaging plane) can vary. Mice imaged with the multi-plane 2-photon microscope can have multiple containers, one for each of the 8 imaging planes recorded by the Multiscope (after data quality review, some of the containers may fail and be removed from the dataset).

Thus, each mouse can have one or more **containers**, each representing a unique imaging plane (**experiment**) that has been targeted across multiple recording **sessions**, under different behavioral and sensory conditions (**session types**).

Illustration of dataset terminology for all ophys data collected from one mouse using multi-plane imaging

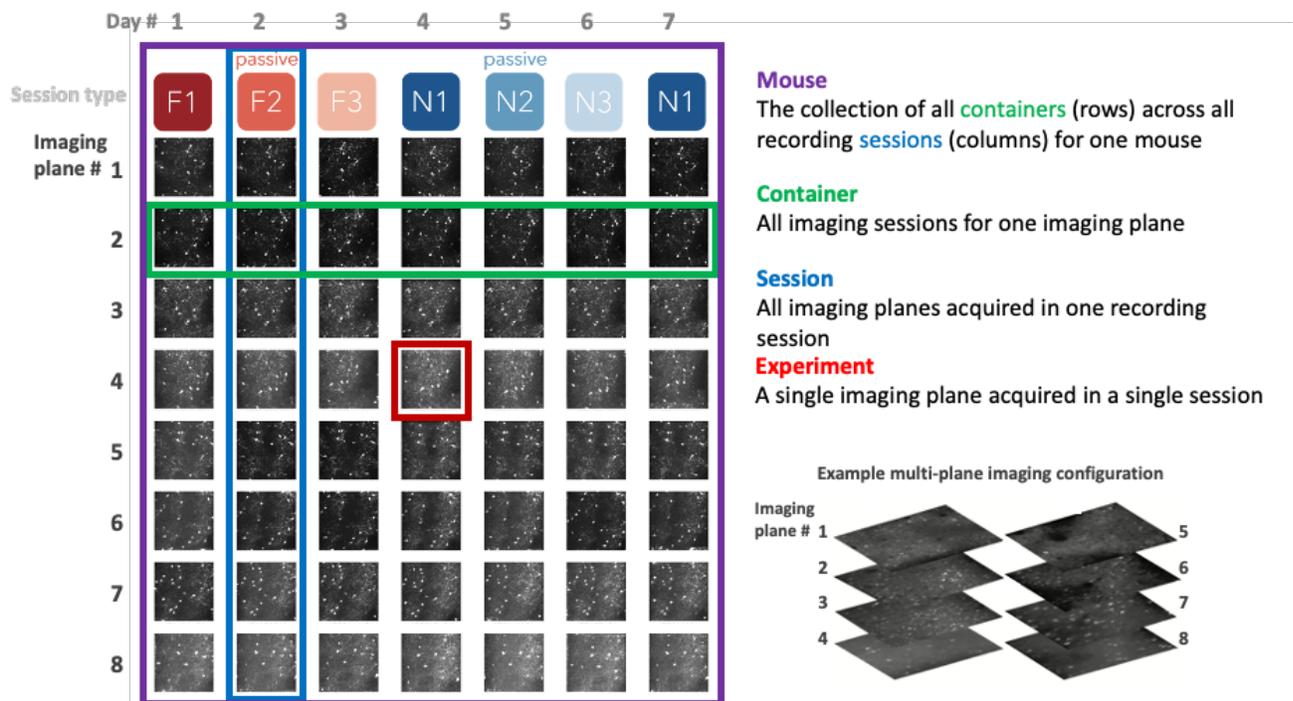


Figure 6. Dataset structure and terminology. Figure represents all the 2-photon imaging data from a single mouse imaged with the multi-plane microscope. Columns are unique recording sessions, with session type indicated as colored boxes in top row. Rows are unique imaging planes, each corresponding to a specific cortical area and depth. Each imaging plane is targeted across multiple recordings sessions. The collection of all sessions for a given imaging plane is referred to as a container. One imaging plane measured in one recording session is referred to as an experiment.

SECTION B: MICE & SURGERY

MICE

All experiments and procedures were performed in accordance with protocols approved by the Allen Institute Animal Care and Use Committee. Male and female transgenic mice expressing GCaMP6 in various Cre-defined cell populations were used in these experiments (Madisen et al., 2015). The four genotypes used in this study were Slc17a7-IRES2-Cre;Camk2a-tTA;Ai93(TITL-GcaMP6f), Slc17a7-IRES2-Cre;Camk2a-tTA;Ai9r(TITL-GcaMP6s) Sst-IRES-Cre;Ai148(TIT2L-GC6f-ICL-tTA2), and Vip-IRES-Cre;Ai148(TIT2L-GC6f-ICL-tTA2). For more information on transgenic mice, see <https://observatory.brain-map.org/visualcoding/transgenic>. Prior to surgery mice were singly-housed and maintained on a reverse 12-hour light cycle (off at 9am, on at 9pm); all experiments were performed during the dark cycle.

SURGERY

Headpost and cranial window surgery was performed on healthy male and female transgenic mice (p37-p63) weighing no less than 15 g at time of surgery and was based on a previously published protocol (de Vries et al., 2020). Pre-operative injections of dexamethasone (3.2 mg/kg, S.C.) and ceftriaxone (100-125 mg/kg, S.C.) were administered 3h before surgery. Mice were initially anesthetized with 5% isoflurane (1-3 min) and placed in a stereotaxic frame (Model# 1900, KOPF; Tujunga, CA), and isoflurane levels were maintained at 1.5-2.5% for surgery. Injections of atropine (0.02-0.05 mg/kg), carprofen (5-10 mg/kg) and Lactated Ringers Solution (LRS) were then administered subcutaneously. An incision was made to remove skin, and the exposed skull was levelled with respect to pitch (bregma-lambda level), roll and yaw. The stereotax was zeroed at lambda using a custom headframe holder equipped with stylus affixed to a clamp-plate (Figure 2a,b). The stylus was then replaced with a custom titanium headframe and well (Figure 1) to center the headframe well at 2.8 mm lateral and 1.3 mm anterior to lambda (Figure 2a, right). The headframe was affixed to the skull with white dental cement (Metabond; Parkell; Edgewood, NY) and once dried, the mouse was placed in a custom clamp to position the skull at a rotated angle of 23° such that the visual cortex was horizontal to facilitate creation of the craniotomy (Figure 2c). A circular piece of skull 5 mm in diameter was manually drilled and removed, and a durotomy was performed. A glass coverslip (cut from a single piece of glass to obtain a “stacked” appearance that consisted of a 5 mm diameter “core” and 7 mm diameter “flange”), was cemented in place with Vetbond (3M; St. Paul, MN). Cement was then applied around the cranial window inside the well to secure the glass window. Mice received post-operative analgesics carprofen (5-10 mg/kg) as well as ceftriaxone 2x/day for 2 days. Post-surgical brain health was documented using a custom photo-documentation system and animals were assessed 1, 2, 7, and 14 days following surgery for overall health (bright, alert and responsive), cranial window clarity and brain health.



Figure 1. Brain Observatory Headframe and well. Isometric view of the custom headframe (manufactured from 1.6 mm (.063 inch) thick titanium 6Al-4V) and light-blocking wells for Scientifica (CENTER) and Mesoscope (RIGHT).

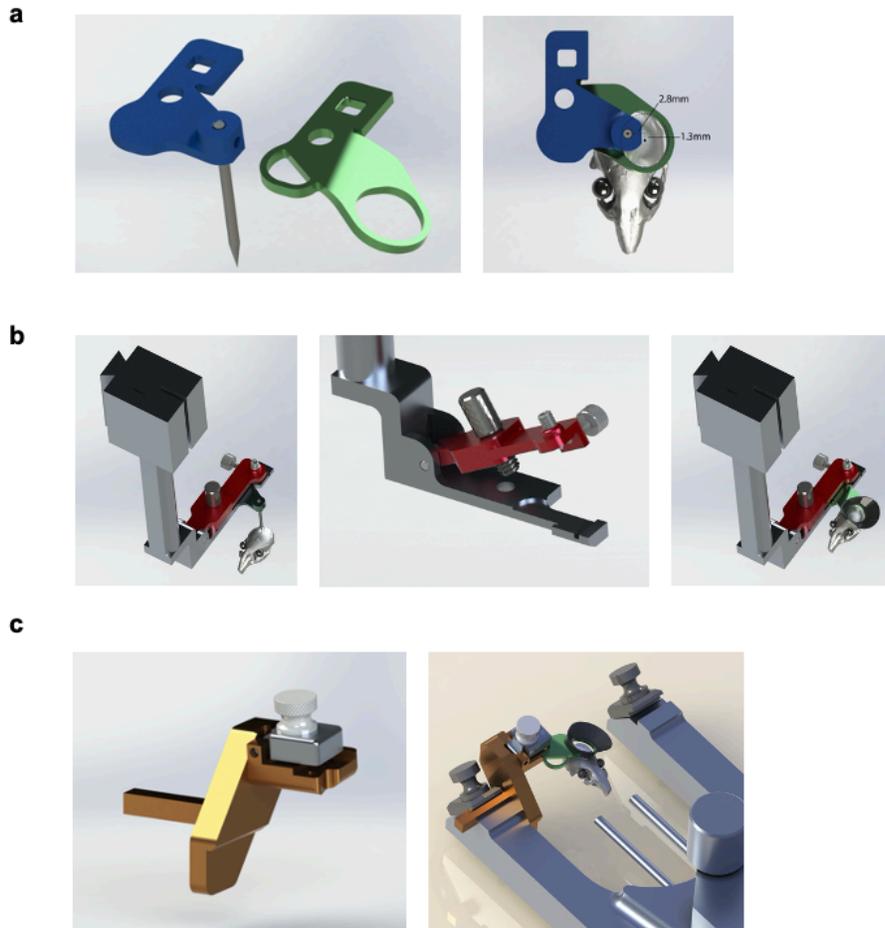


Figure 2. Headframe surgical tooling. a) LEFT: A “stylus” (blue), equipped with a shank identical to that of the headplate (green), is used to locate the mouse skull fiducial, lambda. RIGHT: This stylus places the headplate so that the center of the 10 mm headplate ring is located over the putative location of mouse visual cortex (M/L = -2.8mm, A/P = 1.3mm, with respect to lambda). b) Custom headframe placement tool (“headframe clamp”) is compatible with the KOPF 1900 dovetail interface located on the Z-axis arm and places a clamshell-style clamp (CENTER) parallel to the levelled mouse skull. Once lambda is located with the stylus (LEFT), it is replaced by the headframe (RIGHT) and lowered down along the Z axis to the skull for cementing. c) The cranial window portion of the surgical procedure is facilitated by a custom “levelling clamp” (LEFT) that interfaces with the KOPF 1900 earbar clamp (RIGHT) and pitches the mouse forward 6 degrees. Once the entire earbar apparatus is rotated to 23 degrees, the cranial window plane is positioned perpendicular to gravity. (all images from Groblewski & Sullivan, 2020)

SECTION C: INTRINSIC SIGNAL IMAGING

DATA ACQUISITION

Animal Preparation.

Mice were lightly anesthetized with 1-1.4% isoflurane administered with a Somnosuite (model #715; Kent Scientific, CON) at a flow rate of 100 ml/min supplemented with ~95% O₂ concentrated air (Pureline OC4000; Scivena Scientific, OR). Eye drops (Lacri-Lube Lubricant Eye Ointment; Refresh) were applied to maintain hydration and clarity of eye during anaesthesia. Mice were placed on a lab jack platform and head-fixed for imaging normal to the cranial window. The headframe and clamping mechanism ensured consistent and accurate positioning of the mouse eye in relation to the stimulus screen from experiment to experiment.

Image Acquisition & Processing.

An in-focus image of the surface vasculature was acquired with green LED illumination to provide a fiducial marker reference on the surface of the visual cortex. After defocusing from the surface vasculature (1500 μm along the optical axis), up to 10 independent ISI time series were acquired using red LED illumination to measure the hemodynamic response to visual stimulus-induced brain activity. The resulting images were first processed to maximize the signal-to-noise ratio using time-averaged pixel direct current (DC) signal removal. A Discrete Fourier Transform (DFT) at the stimulus frequency was performed on the pre-processed images. Phase maps were generated by calculating the phase angle of the pre-processed DFT at the stimulus frequency and then used to translate the location of a visual stimulus from the retina to cortical spatial anatomy. A sign map was produced from the phase maps by taking the sine of the angle between the altitude and azimuth map gradients. Averaged sign maps were produced from a minimum of three time series images for a combined minimum average of 30 stimulus sweeps in each direction.

APPARATUS

Image Acquisition System.

The brain surface was illuminated with two independent LED lights: green (peak $\lambda=527\text{nm}$; FWHM=50nm; Cree Inc., C503B-GCN-CY0C0791) and red (peak $\lambda=635\text{nm}$ and FWHM of 20nm; Avago Technologies, HLMP-EG08-Y2000) mounted on the optical lens. A pair of Nikon lenses (Nikon Nikkor 105mm f/2.8, Nikon Nikkor 35mm f/1.4), provided 3.0x magnification ($M=105/35$). The back focal plane of the 50 mm lens was adjacent and coplanar to the cranial window (working distance, 46.5 mm), and was equipped with a bandpass filter (Semrock; FF01-630/92 nm) to selectively allow longer wavelengths of reflected light to reach the camera sensor and to filter screen contamination and ambient light.

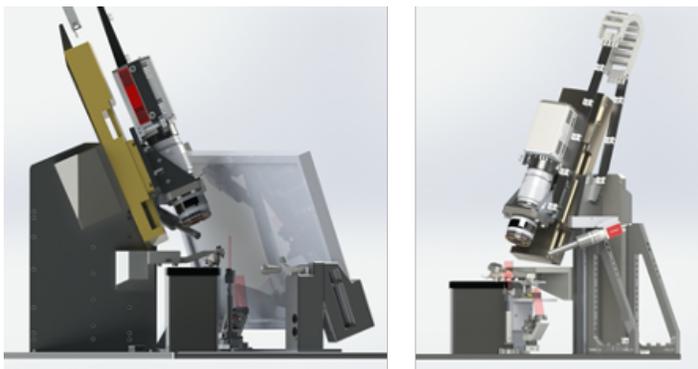


Figure 1. Intrinsic Signal Imaging System.

Illumination and image time series acquisition were controlled with custom software written in Python. An Andor Zyla 5.5 10 tap sCMOS camera was used at a frame rate of 40 Hz with frame intervals timed using the camera's hardware clock running at 40 MHz. Initiation of simultaneous image acquisition and visual stimulus display was hardware triggered from a National Instruments Digital IO board. Frames of 2560 x 2160 pixels and a 16-bit dynamic range were saved to disk at a 4x4 spatial binning and 4x temporal binning, resulting in 640x640 pixels/frame, 10Hz time series, 32-bit dynamic range and a resulting effective pixel size of 10 μm .

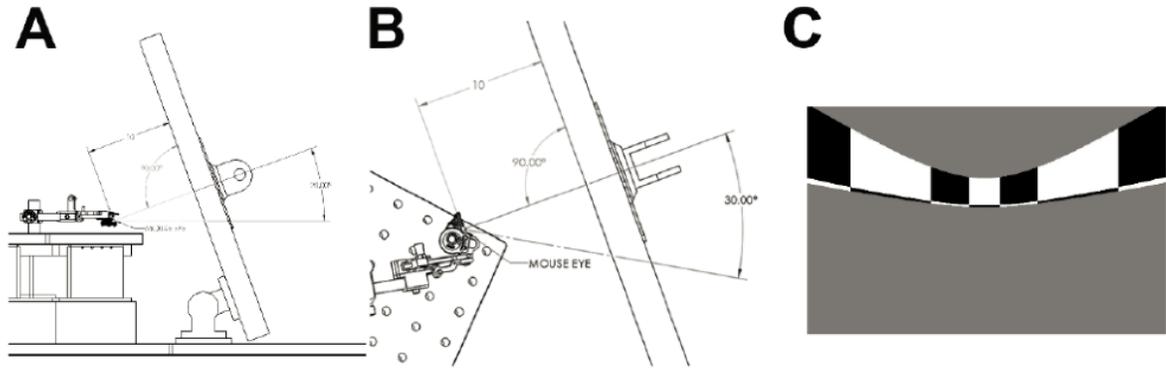


Figure 2. Visual stimulus screen placement and presentation. (A) Side view of the experimental setup showing the center of the monitor positioned at 10 cm away from the eye of the mouse. The monitor was also tilted 70° to the horizontal to present the stimulus parallel to the retina. **(B)** Aerial view of the position of the stimulus monitor showing the screen positioned 30° from the mouse dorsoventral axis also contributing to the parallel relationship between the retina and stimulus display. **(C)** Example snapshot of the stimulus presented, here showing a horizontal sweep moving downwards.

Visual Stimulus for ISI

The lambda-bregma axis of the skull, as positioned in the head frame clamp, was oriented with a 30° pitch relative to horizontal, corresponding to a horizontal eye position ~60° lateral to midline and a vertical position ~20° above the horizon (Oommen & Stahl, 2008). To ensure maximal coverage of the field of view, a 24" monitor was positioned 10 cm from the right eye. The monitor was rotated 30° relative to the animal's dorsoventral axis and tilted 70° off the horizon to ensure that the stimulus was perpendicular to the optic axis of the eye (Figure 2A & B).

The visual stimulus displayed was comprised of a drifting bar containing a checkerboard pattern, alternating black and white as it sweeps along a gray background (Figure 2C). The stimulus bar sweeps across the four cardinal axes 10 times in each direction at a rate of 0.1 Hz (Kalatsky and Stryker, 2003). The drifting bar measures 20° by 155°, with individual square sizes measuring at 25°. The stimulus was warped spatially so that a spherical representation could be displayed on a flat monitor (Marshel et al., 2011).

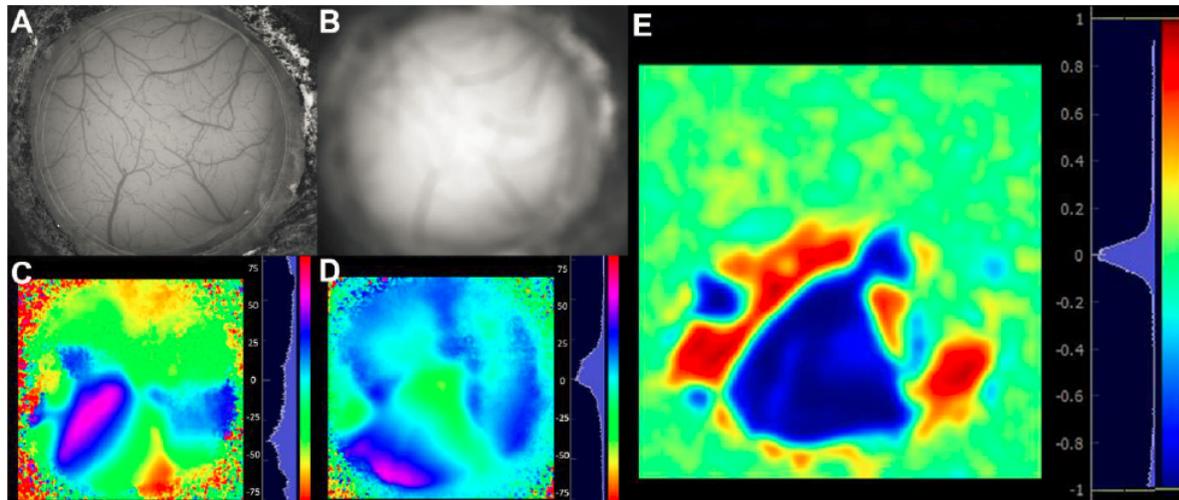


Figure 3. Images and maps generated from an individual time series images during an ISI session. A) Image of the vasculature at focal plane, and B) defocused image were used as fiduciary markers to provide cortical reference. For each trial, consisting of 10 sweeps of the drifting bar in the four cardinal directions, five maps were computed: azimuth (C) and altitude (D) maps, amplitude maps of the signalscale bar in degrees. (D) Altitude map computed from vertical sweeps of the drifting bar stimulus, with scale bar in degrees. (E) Visual sign map showing area boundaries, generated from the sine of the angle between the horizontal and vertical map gradients

Automated Sign Map Segmentation and Annotation.

An automated segmentation and annotation module was developed to delineate boundaries between visual areas. For each experiment, a visual field sign map was computed as the sine of the angle between the horizontal and vertical map gradients. Negative values (visualized in blue in Figure 3E) represent a mirror image of the visual field while positive values (red) represent non-mirror image. The sign map is automatically segmented into distinct visual areas with an algorithm (based on algorithm in Garrett et al, 2014) that used three criteria to segregate visual areas: 1) each area must contain the same visual field sign; 2) each area cannot have a redundant representation of visual space; 3) adjacent areas of the same visual field sign must have a redundant representation. To facilitate automated annotation, a set of 35 ISI experiments generated prior to the collection optical physiology data were used to build an initial map of 12 major cortical visual areas (VISp, VISpl, VISl, VISli, VISal, VISlla, VISrl, VISam, VISpm, VISmma, VISmmp, VISm). The identity of each area was first manually annotated to provide “ground truth” identification (Figure 4). Each sign map was aligned to a canonical space by shifting the center of mass of the primary visual cortex, V1, to the origin, and rotating the image so that the direction of the altitude retinotopy gradient of primary visual area was horizontal and consistent across experiments. Statistics for area sign, location, size and spatial relationships were compiled.

Subsequent ISI datasets were automatically segmented, and the automated annotation algorithm compared the sign, location, size, and spatial relationships of the segmented areas against those compiled in the ISI-derived atlas of visual areas. A cost function, defined by the discrepancy between the properties of the matched areas, was minimized to identify the best match between visual areas in the experimental sign map and those in the atlas, resulting in an auto-segmented and annotated map for each experiment. The automated ISI segmentation and annotation modules achieved ~97% accuracy on a pilot dataset. Manual correction and editing of the results included merging and splitting of segmented and annotated areas to correct errors.

Eccentricity and Target Map Generation.

Two eccentricity maps were generated (Figure 4A-E). The first is a map of eccentricity from the center of visual space, at the intersection of 0° altitude and 0° azimuth (Figure 4D). If the retina is centered on the origin of the visual stimulus at the center of the monitor, the center of visual space (in stimulus coordinates of altitude and azimuth) should fall approximately on the anatomical center of V1, corresponding to the center of the retina. For example, if the optic axis of the eye was pointed at the upper portion of the screen rather than the center of the stimulus (the origin, 0° altitude and azimuth in screen coordinates), the values of the altitude map would be

shifted upwards, resulting in a corresponding shift in the origin of the map of visual eccentricity away from V1's center. The value of retinotopic eccentricity at the V1 centroid was used as a QC criterion, as described below, to identify experiments where the eye significantly deviated from the center of the stimulus ($>15^\circ$ shift).

The maps of altitude and azimuth represent a mapping of the screen onto the retina. Accordingly, the exact values and range of the maps vary across experiments as a result of differences in eye position relative to the monitor. To provide a reliable map for subsequent targeting of 2-photon calcium imaging experiments, a consistent anatomical coordinate corresponding to the center of V1 (which maps to center of the retina) was used to realign the maps. A map of eccentricity from the V1 centroid was produced by shifting the origin of the map of visual eccentricity (Figure 4D) to the coordinates at the V1 centroid, showing the retinotopic gradients relative to this point (Figure 4E). A representation of the corresponding retinotopic location is present in nearly all higher visual areas (Figure 4E, indicated in red in the heat map). Using this location as the target for optical physiology experiments ensures that recorded neurons represent a consistent region on the retina, approximately at the center of the right visual hemifield.

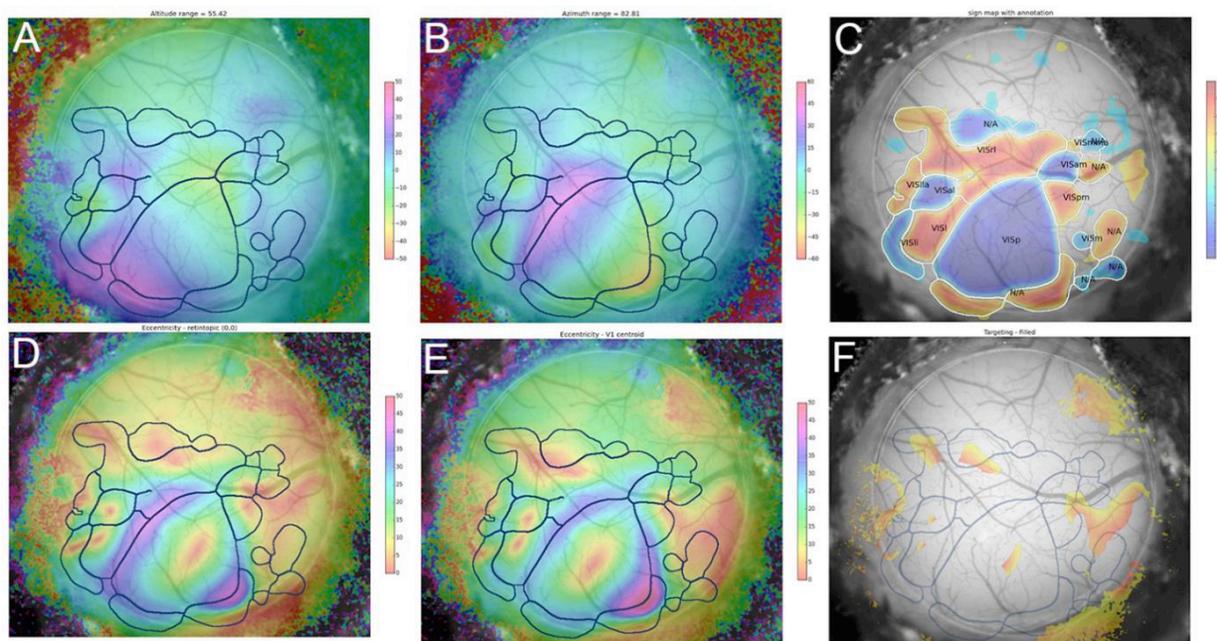


Figure 4. Images and averaged maps generated post-acquisition of an ISI session. All selected trials are averaged to produce azimuth (A) and altitude (B) maps, and a map of visual field sign with segmented area boundaries overlaid on a vasculature image (C). From the altitude and azimuth maps and segmented area boundaries, a map of visual eccentricity (D), a map of eccentricity from the V1 centroid (E), and a target map (F) are computed. List of acronyms: Primary Visual Area (VISp), Posterolateral visual area (VISpl), Laterointermediate area (VISli), Lateral visual area (VISl), Anteromedial visual area (VISal), Laterolateral anterior visual area (VISlla), Rostrolateral visual area (VISrl), Anteromedial visual area (VISam), Posteromedial visual area (VISpm), Medial visual area (VISm), Mediomedial anterior visual area (VISmma), Mediomedial posterior visual area (VISmp). Scale bar in degrees for A, B, D & E.

To provide a discrete target location for subsequent 2-photon imaging, targeting maps were created from the maps of eccentricity at the center of V1 (Figure 4E) by restricting the map to values of eccentricity that are within 10 degrees from the origin. In addition, the target map was limited to retinotopic values that are negative for both altitude and azimuth. This gives the target a specific directionality, displaying values only for the lower peripheral quadrant of visual space to bias the targeting closer to the center of the 2-photon system's visual display monitor. This targeting map (Figure 4F) was overlaid on an image of the surface vasculature to provide fiducial landmarks to guide optical physiology recording sessions and to ensure that the imaged locations were retinotopically matched across areas.

The quality control process for the ISI-derived maps included four distinct inspection steps (Figure 5):

1. The brain surface and vasculature images were inspected post-acquisition for clarity, focus, and position of the cranial window within the field of view (QC-1). The vasculature images were also examined for the presence of white artifacts on the brain surface, an indicator of potential brain damage, was grounds for failing the mouse out of the pipeline.
2. Individual trials were inspected for visual coverage range and continuity of phase maps, localization of the signal from the amplitude maps and stereotypical organization of sign maps. Only trials respecting these criteria were included in the final average, and a minimum of 3 trials were required (QC-2).
3. Visual area boundaries were delineated using automated segmentation, and maps were curated based on stringent criteria to ensure data quality. The automated segmentation and identification of a minimum of six visual areas including VISp, VISIm, VISrI, VISaI, VISaM and VISpm was required. A maximum of three manual adjustments were permitted to compensate for algorithm inefficiency (QC-3).
4. Each processed retinotopic map was inspected for coverage range (35-60° altitude and 60-100° azimuth), bias (absolute value of the difference between max and min of altitude or azimuth range; <10°), alignment of the center of retinotopic eccentricity with the centroid of V1 (<15° apart), and the area size of V1 (>2.8 cm²).

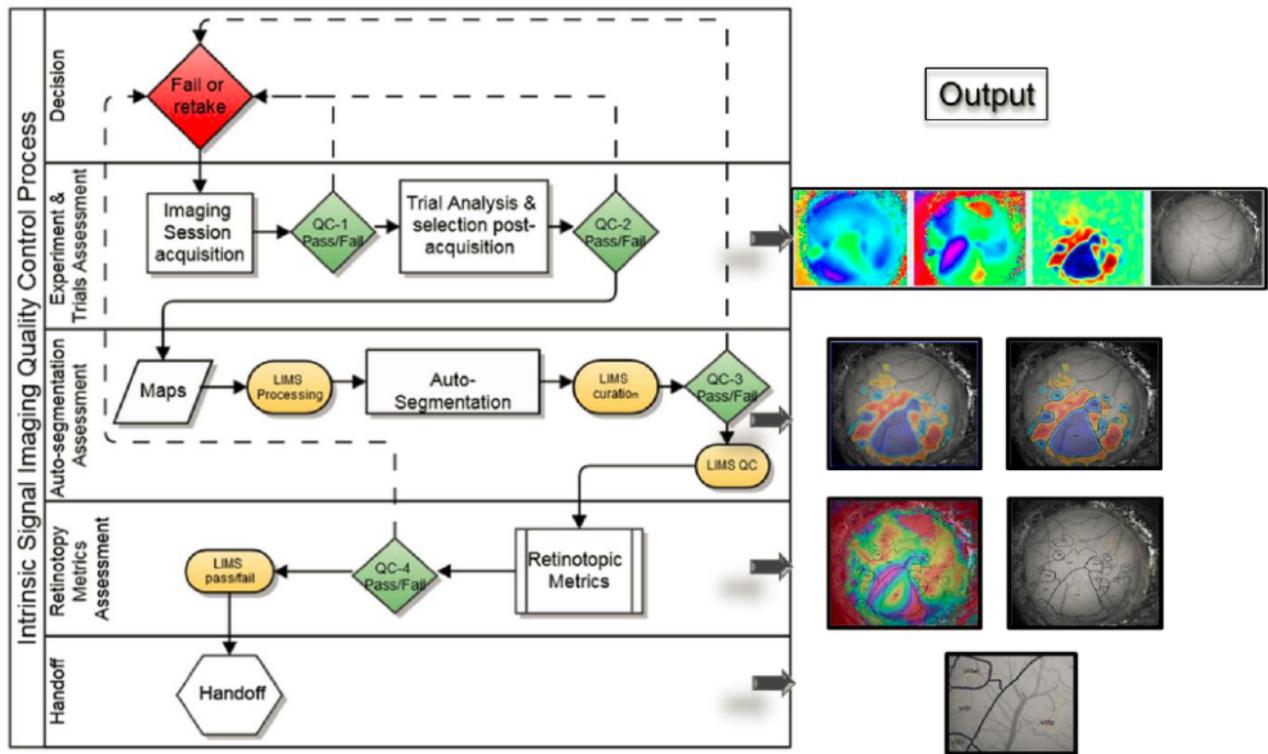


Figure 5. Schematic of ISI process and quality control. Beginning at “Imaging session acquisition”, a number of process and QC steps were followed, each resulting in the maps shown to the right.

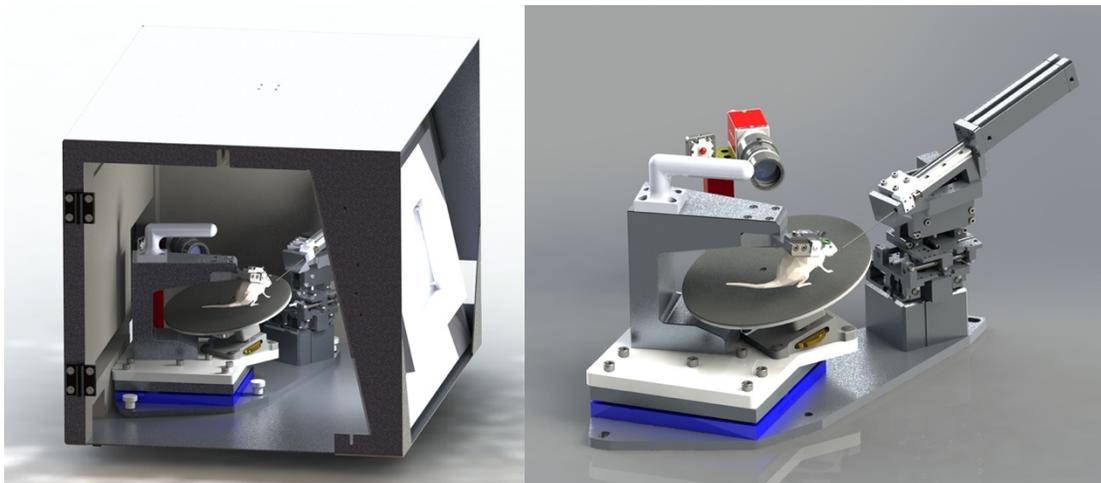


Figure 2. Behavior enclosure and stage. Custom behavior enclosure (LEFT) and removable behavior stage, shown with lick spout module and camera (RIGHT).

CHANGE DETECTION TASK

Overview: Mice were trained for 1 hour/day, 5 days/week using a behavioral program implementing a go/no-go change detection task schematized in Figure 1. Briefly, mice were trained to lick a reward spout when the identity of a flashed visual stimulus changed identity. If mice responded correctly within a short, post-change response window (150-750ms before compensating for monitor display lag) a water reward was delivered. The volume of contingent rewards was 10uL in Stages 1 and 2, and reduced to 7uL after the first 3 sessions of Stage 3. The four stages of the training protocol are shown in Table 1

Stage	Stimulus	Stimulus Presentation	Response (ms)	Window	Contingent Rewards	Duration (min)
0	Square-wave gratings	Static	NA		False	15
1	Square-wave gratings	Static	1000		True	60
2	Square-wave gratings	250 ms stimulus; 500 ms gray period	600		True	60
3	Natural Images	250 ms stimulus; 500 ms gray period	600		True	60

Table 1. Behavior training stages

On Day 1 of the automated training protocol (shown in Figure 3) mice received a short, 15-min “open loop” session during which non-contingent water rewards were delivered coincident with 90° changes in orientation of a full-field, static square-wave grating (Stage 0). This session was intended to 1) introduce the mouse to the fluid delivery system and, 2) provide the technician an opportunity to identify the optimal lick spout position for each mouse. Each session thereafter was run in “closed loop”, and progressed through 3 stages of the operant task: 1) static, full-field square wave gratings (oriented at 0° and 90°, with the black/white transition always centered on the screen and the phase chosen randomly on every trial), 2) flashed, full-field square-wave gratings (0° and 90°, with phase as described in 1), and 3) flashed full-field natural scenes (8 natural images used in the *Allen Brain Observatory*, <http://observatory.brain-map.org/visualcoding>). Once in the final stage of the task, mice were introduced to the additional session components that would be included during imaging sessions (5 min of gray screen immediately before and after the behavior session, as well as the final 5 min movie).

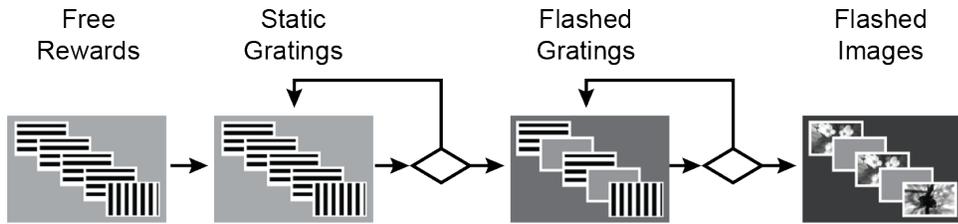


Figure 3. Automated training of change detection task. Schematic showing progression of training stages.

Progression through training stages: Starting with Stage 1, the advancement criteria required mice to achieve a session maximum performance of at least d' -prime=2 (calculated over a rolling 100 trial window without trial count correction) during two of the last 3 sessions. The fastest progression from Stage 1 to Stage 3 was four training days.

Behavior session and trial structure: Each behavior session consisted of a continuous series of trials, schematized below.

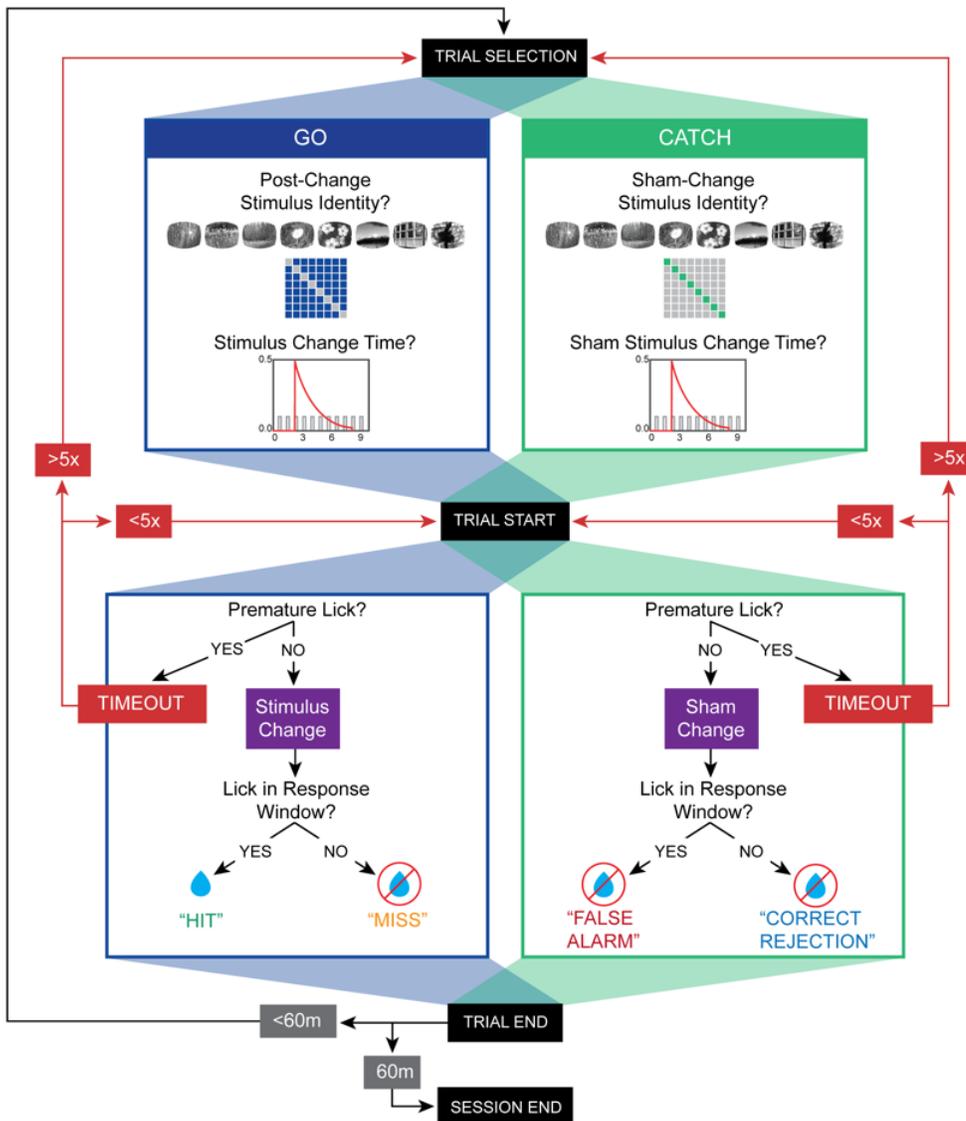


Figure 4. Task flow diagram. The “Flashed Images” stage of the change detection task consists of 8 images, resulting in 64 possible image transitions, including both GO and CATCH trials. GO trials comprise 87.5% of all trials and are represented in the off-diagonal portions of the 8x8 change matrix. CATCH trials comprise 12.5% of all trials and are represented in the diagonal of the matrix. Each trial was first selected as either GO or CATCH and a post-change (or sham-change) image identity was chosen from the change matrix. The stimulus change (or sham-change) time was then selected from a truncated exponential distribution between 2.25s to 8.25s. As stimuli are presented every 750 ms, the actual change time was determined as the nearest flash from the drawn time. Once a trial started, a premature lick (i.e., a lick that occurred prior to the predetermined change time) resulted in a 300 ms timeout and the trial was restarted at the time of the next scheduled stimulus presentation. If an animal caused a trial to timeout 5 times, new trial parameters were selected. If no premature licks were recorded, the trial progressed and the stimulus change occurred at the predetermined change-time. On GO trials, a lick detected within response window (150 ms to 750 ms following image change, before compensating for display lag) resulted in a “HIT” (and subsequent reward delivered) whereas a lack of response resulted in a “MISS”. On CATCH trials, a lick within the window following the sham-change resulted in a “FALSE ALARM”, whereas a lack of response resulted in a “CORRECT REJECTION”. Following the stimulus change and response window the trial ended and a new trial was selected. The session ended after 60 minutes.

Prior to the start of each trial a trial-type and change-time were selected. Trial-type was chosen based on predetermined frequencies such that “GO” and “CATCH” trials occurred with specified probabilities. In stages 1 and 2, the catch probability was ~36%. In stages 3 and above, the catch probability was initially ~30%. However, later sessions implemented a matrix sampling algorithm that ensured that each image transition was sampled equally, pushing the actual catch probability to ~12.5%. Change-times were selected from a truncated exponential distribution ranging from 2.25 to 8.25 seconds (mean of 4.25 seconds) following the start of a trial. Due to computational lag when aligning change-time with a stimulus flash, the actual distribution of change times was shifted to the right by one 750ms flash cycle (with only a small fraction of changes occurring at 2.25 seconds) resulting in a mean change time of 4.2 seconds. In trials when a mouse licked prior to the stimulus change the trial was reset, and a timeout period was imposed. The number of times a trial could be reset before re-drawing the timing parameter was limited to five. In all, this trial structure leads to a sampling of “GO” and “CATCH” trials, that when combined with mouse responding, yields “HIT”, “MISS”, “FALSE ALARM”, and “CORRECT REJECTION” trials.

In addition to the four trial types described above, behavior sessions contained a subset of “free reward” trials (“GO” trials followed immediately by delivery of a non-contingent reward). Behavior sessions across all phases began with 5 “free-reward” trials. Additionally, in order to promote continued task performance throughout the behavior session “free reward” trials were delivered after 10 consecutive “MISS” trials. All non-contingent rewards were 5uL in volume.

Transition to Imaging: Mice were deemed to be “transition ready” when they had achieved the following performance criteria during the final phase of behavior:

1. A peak d-prime of >1 for three consecutive sessions
2. At least 100 HIT and/or MISS trials on three consecutive sessions

If at any time during the final stage of behavior mice did not meet these criteria they would become “lapsed” until they met the criteria and returned to “transition ready”. Additionally, due to operational constraints, mice were sometimes not immediately transitioned to the imaging phase of the experiment despite having met the criteria.

BEHAVIOR METRICS

Hit and false alarm rates: The hit rate was calculated as the fraction of go-trials in which the mouse licked in a 0.150 to 0.750 second window following the non-display-lag-compensated image display time. On Catch trials, a response window was defined as a 0.150 to 0.750 second window following the sham change. False alarm rates were calculated as the fraction of catch-trials in which animal emitted a lick in this response window.

Reaction times were recalculated post-hoc after accounting for the calculated display lag, which ranged from ~20-35 ms. Due to variability in the display lag calculations, the minimum and maximum reaction times that resulted in trial classification (hit/miss/false-alarm/correct-rejection) will vary from session to session.

Hit and false alarm rates were corrected to account for low trial counts by bounding their values using the following formula (Macmillan and Creelman, 2004):

$$\frac{1}{2N_H} \leq R_H \leq 1 - \frac{1}{2N_H}$$

$$\frac{1}{2N_F} \leq R_F \leq 1 - \frac{1}{2N_F}$$

Where R_H and R_F represent the hit and false alarm rates, and N_H and N_F represents the number of hit and false alarm trials.

D-prime (d'), which is a measure of the relative difference in response probabilities across the two trial types, is defined as:

$$d' = Z(R_H) - Z(R_F)$$

in which Z represents the inverse cumulative normal distribution function.

The AllenSDK returns hit rates, false alarm rates, and d-prime calculated over a rolling 100 trial window, excluding aborted trials (trials where the animal responded before the stimulus change). The reward rate is calculated over a 25 trial rolling window and provides a measure of the rewards earned per unit time (in units of rewards/minute). This provides a rough proxy for task engagement, with periods of high reward rates indicating engagement in the visual task. Low reward rates can indicate either disengagement (lack of responses), or extremely high rates of anticipatory licking (thereby continually delaying stimulus changes). In cases where the SDK has labeled epochs as either 'engaged' or 'disengaged', this corresponds to reward rates above and below 2 rewards per minute.

See detailed documentation of the SDK methods in the [`get_rolling_performance_df`](#) function in the `behavior_session` module, the [`calculate_reward_rate`](#) method in the `trials_processing` module, and in the [readme](#).

RUNNING SPEED CALCULATION

The animal's running speed was monitored using an analog encoder mounted at the center of the 6.5" (16.5 cm) diameter flat running wheel (see section APPARATUS for details). The encoder was provided with a continuous 5V reference voltage (v_{in}). A reference signal (v_{sig}), along with the actual value of v_{in} , was read from the encoder at discrete sampling intervals and logged to the data file. The ratio of v_{sig} to v_{in} corresponded to the position of the encoder relative to a zero-point. For example, $v_{sig}/v_{in} = 0$ corresponds to 0 degrees, $v_{sig}/v_{in} = 0.5$ corresponds to 180 degrees, etc. As the encoder crossed its zero-point ($v_{sig}/v_{in} = 1.0$), v_{sig} returned to 0. The rotational speed of the encoder was then accessible as the change in position since the last encoder measurement divided by the time interval since the last measurement (i.e., the derivative). The rotational speed was translated to a linear speed under the assumption that the mouse was running at a location 2/3 of the distance from the wheel center.

In practice, there were a few sources of error in the recordings including inconsistencies in the timing of the analog channel reads, spurious transients on the analog input channels, and inconsistencies in the derivative as the encoder crossed the zero-point. The following processing steps were followed to provide an estimate of the running speed despite these issues. All code for encoder processing and running speed calculation is visible in the [brain_observatory.behavior.running_processing](#) module of the AllenSDK.

The value of v_{sig} was first 'unwrapped', to account for the crossings of the zero-point in the encoder and facilitate the derivative step to follow. This was accomplished by first removing any voltage values greater than

5.1 (to remove spurious transients), then identifying and compensating for the zero-crossing points (or ‘wraps’). Below is a 10 second sample of raw encoder data before (top) and after (bottom) unwrapping

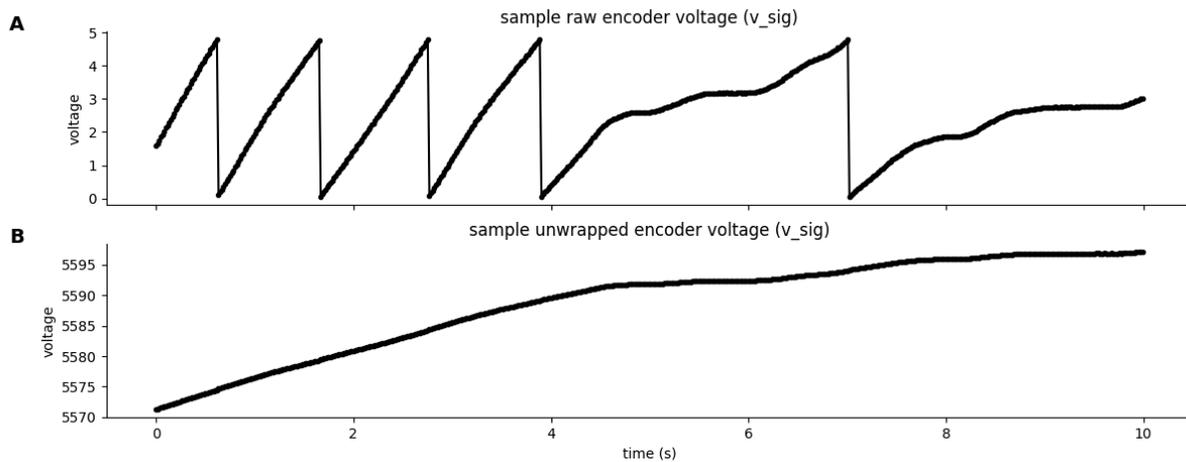


Figure 5. A) raw voltage signal (*v_sig*) from the analog encoder. The values range between 0 and a 5 V reference voltage. B) The encoder voltage after the ‘unwrapping’ process.

The voltage was converted to angular speed, in radians/s, by differentiating the unwrapped *v_sig* values, dividing by *v_in*, then multiplying by 2π . Angular speed was converted to linear speed using the above-described assumption of the animal’s running position on the wheel as follows:

$$\text{running_speed_cm_per_sec} = \text{angular_speed} * (2/3 * \text{wheel_radius})$$

Remaining transients at the wrap points were identified and removed by limiting the values at the wrap points to be within the range defined by all other data points in a +/- 0.25 second window. Any additional transients with z-score of 10 or greater were removed (set to NaN). Finally, inconsistencies in the analog channel read times were accounted for by smoothing the raw running speed with a 10 Hz lowpass Butterworth filter. Both the unfiltered (but wrap and transient corrected) and the filtered running speeds are available to end users as, respectively, the ‘*running_speed_raw*’ and ‘*running_speed*’ attributes. Below is the linear speed, in cm/s, for the same 10 second period shown above in Figure 5:

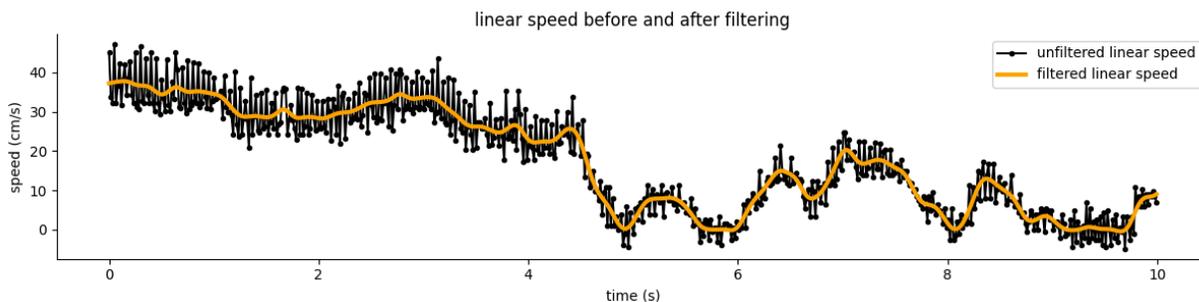


Figure 6. Linear running speed calculated from the sample voltage signal shown in Figure 5. The unfiltered signal is shown in black. The result of filtering this signal with a 10 Hz lowpass Butterworth filter is shown in orange. The filtered signal is the default running signal provided by the AllenSDK.

SECTION E: IN VIVO 2-PHOTON CALCIUM IMAGING

HARDWARE & INSTRUMENTATION

Single-plane calcium imaging was performed using a 2-photon microscope (Scientifica Vivoscope), as used by de Vries et al., 2020. Scientifica microscope design is based on 8 kHz resonant scanning mirror and employs conventional hardware (photomultiplier tubes, Hamamatsu; transimpedance amplifier, Femto; DAQ hardware, National Instruments; 16x imaging objective, Nikon) to collect emitted fluorescence and form an image on the acquisition computer. The microscope is controlled by the company's proprietary LABView software SciScan. Laser excitation was provided by a Ti:Sapphire laser (Chameleon Vision, Coherent) at 910 nm. Pulse dispersion compensation was set at $\sim 10,000$ fs². Movies were recorded at 30Hz using resonant scanners over a 400 μ m field of view.

Multi-plane calcium imaging was performed using a Dual-Beam Mesoscope (Multiscope) that allowed us to double imaging throughput (Orlova, Tsyboulski, Najafi et al., 2020). The second laser beam was introduced to the original 2P-RAM system (Sofroniew et al., 2016), packaged into a compact opto-mechanical add-on unit, and optimized for ease of alignment. The dual-beam modification consisted of a 1) delay line used to split the original laser beam into 2 and delay one of the beams by half a period of the excitation laser; 2) a secondary z-scanner and 3) custom-built demultiplexing unit. The delay line allowed for temporal encoding of the excitation beam and further demultiplexing of the detected fluorescence based on the arrival time at the photodetector. A secondary z-scanner allowed us to send two beams to the two focal planes located along Z axis. Laser excitation was provided by a Ti:Sapphire ultrafast laser (Chameleon Ultra II, Coherent). Pulse dispersion compensation was optimized for GCaMP6 using a custom-built external pulse compensation module based on a single-prism four-path design (Sofroniew et al., 2016; Akturk, et al., 2006). The Multiscope was controlled with customized ScanImage software (VidrioTech) as well as an in-house developed Workflow Sequencing Engine (described below). Like conventional 2-photon microscopes, the emitted fluorescence was detected using a single photomultiplier tube, and a custom analog demultiplexing circuit was used to separate fluorescence from two planes (Orlova, Tsyboulski, Najafi et al., 2020). This was achieved by multiplying the PMT signal with two complimentary square waveforms, where each waveform corresponded to the temporal window during which fluorescence received by the PMT consisted of the signal from one of the focal planes. The duration of the integration window was 6.25 ns (half a period of the excitation laser's pulse train), which is not enough to fully capture the decay of fluorescence, which resulted in the tail of fluorescence leaking to the opposite integration window and causing inter-plane crosstalk ($\sim 10\%$ remaining crosstalk on average). We used an ICA-based demixing algorithm (described below) to further clean up the data acquired in simultaneously imaged focal planes. National Instruments data acquisition hardware (PXI chassis, PXIe6363 DAQ boards) was used to control the microscope, form, and record the image.

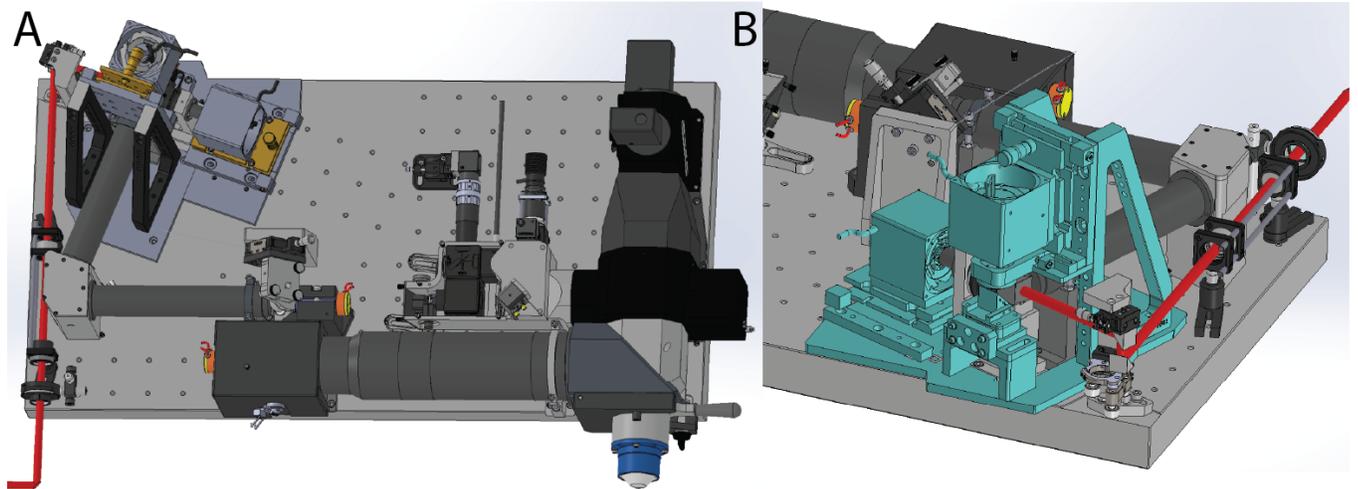


Figure 1. Dual-Beam Mesoscope (Multiscope), (A), Zoom-in to the Dual-beam add-on to MesoScope (B)

VISUAL STIMULATION DURING IN VIVO 2-PHOTON IMAGING

The offset of the screen from the eye was 15 cm, with the screen center positioned 118.6 mm lateral, 86.2 mm anterior and 31.6 mm dorsal to the right eye, corresponding with the average gaze direction with the screen placed normal to the gaze axis (Oommen & Stahl 2008), and the long axis of the screen parallel to the optical table (Figure 2).

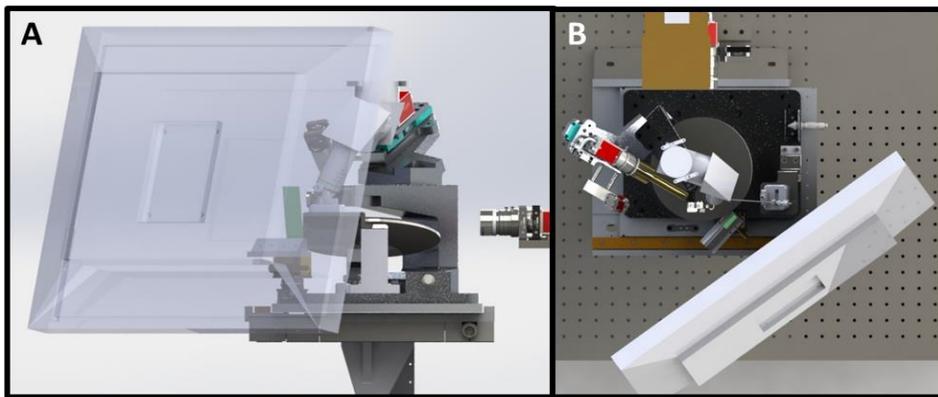


Figure 2. Front (A) and top (B) view of the revised stimulation screen and the imaging system.

The screen spanned $120^\circ \times 95^\circ$ of visual space without accounting for stimulus warping. Each screen (ASUS PA248Q LCD monitor) was gamma calibrated using a USB-650 Red Tide Spectrometer (Ocean Optics). Luminance was measured using a SpectroCAL MKII Spectroradiometer (Cambridge Research Systems). Monitor's brightness (30%) and contrast (50%) corresponded to a mean luminance of 50 cd/m^2 .

MOUSE BODY AND EYE MONITORING

During calcium imaging experiments, eye movements and animal posture were recorded (Figure 3A). The left side of each mouse was imaged with the stimulation screen in the background (Figure 3B) to provide a detailed record of animal behavior during the experiment. The eye facing the stimulus monitor (right eye) was recorded using a custom IR imaging system (Figure 3C). No pupillary reflex was evoked by any of these illumination LEDs.

Behavior Monitoring Hardware	
Camera	Allied Vision, Mako G-032B - GigE interface 30 fps. 33 ms exposure time, camera hardware gain: 10
Illumination	LED Engine Inc., LZ4-40R308-0000 - 740 nm
Camera Lens	Thorlabs MVL8M23, 8 mm EFL, f/1.4, for 2/3" C-Mount Format Cameras, with lock
Filters	Camera: a 785 nm short-pass filter (Semrock, BSP01-785R-25) suppressed light from the eyetracking LED. Illumination: a 747+/-33 nm bandpass filter (Thorlabs, LB1092-B-ML) in front of the LED prevented visible portion of the LED spectrum from reaching the mouse eye.
Eye Monitoring Hardware	
Camera	Allied Vision, Mako G-032B with a GigE interface, used at 30 fps, 33 ms exposure time, camera hardware gain of 10-20
Illumination	LED Engin Inc., LZ1-10R602-0000 (850 nm) Lens in front of the LED : Thorlabs, LB1092-B-ML
Camera Lens	Infinity, InfiniStix 130 mm working distance, 0.73x magnification
Filters	Camera: 850+/-10 nm single-band bandpass filter (Semrock FF01-850/10-25) Dichroic (used to fold beam path): Semrock, FF750-SDi02-25x36

Table 1. Hardware for behavior and eye monitoring.

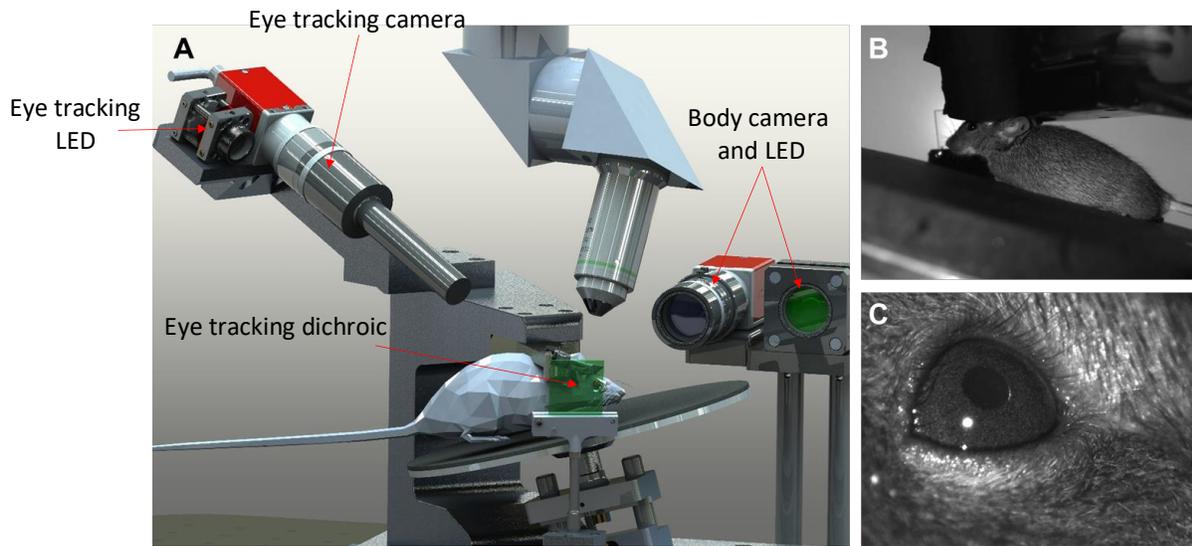


Figure 3. Geometry of body and eye monitoring hardware. Schematic shown in (A); sample images from body camera (B), and eye tracking camera (C).

DATA SYNCHRONIZATION

Temporal synchronization of all data-streams (calcium imaging, visual stimulation, body and eye tracking cameras) was achieved by recording all experimental clocks on a single NI PCI-6612 digital IO board at 100 kHz (Figure 4). All input lines on the synchronization board were sampled simultaneously as a single 32-bit integer at 100 kHz (hardware-timed). To mitigate data size after collection, only changing events and their corresponding sample numbers were stored on a dedicated HDF5 file along with additional metadata.

WORKFLOW SEQUENCING ENGINE (WSE)

The Workflow Sequence Engine (WSE, schematized in Figure 4) is a custom software designed to synchronize acquisition of behavior, 2-photon imaging, eye-tracking, body-tracking, and to automate the experimental workflow when possible. The WSE used a distributed messaging interface and directly interfaced with the microscope software (Scientifica SciScan and ScanImage), the visual stimulation computer, the synchronization computer, as well as the body and eye-tracking computer. During experimental steps that required user intervention, the WSE prompted the rig operator to perform and acknowledge completion of manual tasks. Finally, the WSE coordinated data transfer to a centralized data repository.

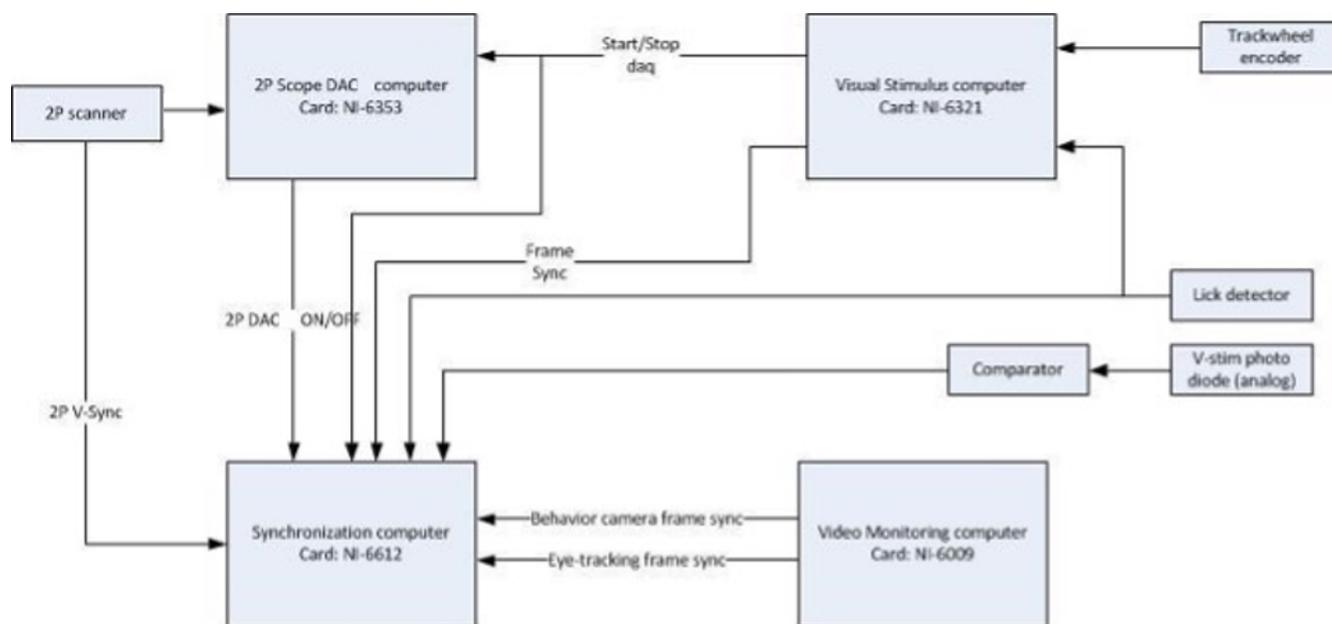


Figure 4. Schematic diagram of the data synchronization architecture for 2-photon imaging experiments.

DATA COLLECTION

All setup of the 2-photon imaging experiments was conducted under ambient red light to maintain the reversed day-night cycle, and imaging itself was performed in the dark. Mice were head-fixed on top of a rotating disk and free to walk or run at will. The disk was covered with a layer of removable foam (Super-Resilient Foam, 86375K242, McMaster) to minimize motion-induced artifacts during imaging sessions.

On the first day of imaging at a new location, the targeting map (generated in ISI, Section D, Figure 4F) was used to select spatial coordinates. The microscope objective was moved to position the imaging objective above the target location. Comparison of superficial vessel patterns were used to verify the appropriate location by imaging over a field of view of $\sim 800 \mu\text{m}$ using epi-fluorescence and blue light illumination for Scientifica microscopes, or the full 5mm field of view for Multiscope. Once a cortical region was selected, the objective was shielded from stray light coming from the stimulation screen using black tape. In 2-photon imaging mode, the desired depth of imaging was set relative to the cortical surface. On subsequent imaging days, for **single-plane experiments recorded** on Scientifica microscopes, a return to the same location was achieved using three alignment steps (Figure 5):

1. The pattern of vessels was matched to those from the previous imaging day in epifluorescence mode using a red-green overlay of live acquisition with a reference image from the previous day.

2. Using the overlay, the same vessels were matched in 2-photon mode using a surface image acquired in the previous session to provide adequate coverage of the 2-photon field of view ($\sim 400 \mu\text{m}$).
3. Once at depth, the current image was matched with a previous depth image using fine stage movements ($\pm 5 \mu\text{m}$) and grid overlay ($100 \mu\text{m}$ pitch) displayed on the reference and target live images.

For **multi-plane experiments** recorded with Multiscope, cell-matching was achieved via a similar procedure, with a few key differences. First, the full cranial window is visualized in epi-fluorescent mode and a vasculature image is taken. Then, the full FOV is imaged in 2p-mode, and the same area at the surface of the brain was located using the 2p-surface images acquired during the initial experiment in the series (parent experiment). Then, the field of view at the target imaging depth was matched to the image of the field of view acquired during the parent experiment by moving up and down in the z-dimension until cells and other image features were appropriately matched.

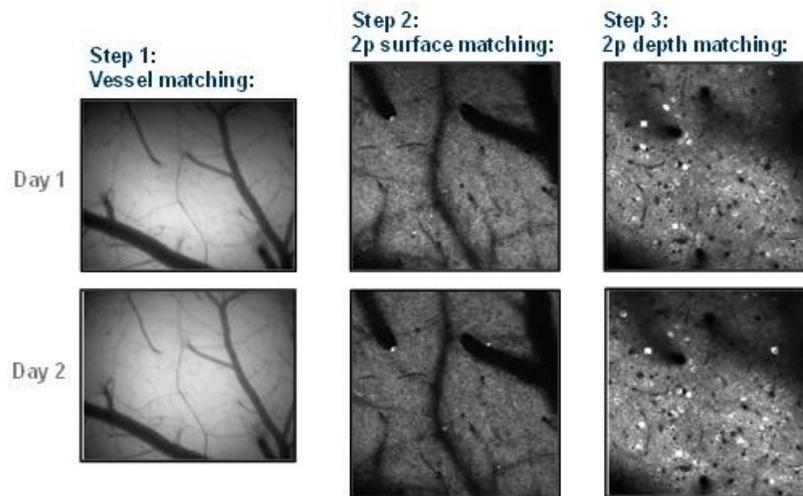


Figure 5. Illustration of the alignment procedure to record from the same population on different days, using surface vasculature for field of view alignment.

Once a depth location was stabilized, a combination of PMT gain and laser power was selected to maximize image signal-to-noise and dynamic range while avoiding pixel saturation (max number of saturated pixels < 1000). A power lookup table (Table 2) was used to guide rig operators through this process. For multi-plane experiments, the SNR of coupled planes was equalized by controlling how much power was being sent towards each beam while monitoring pixel histogram of both channels. Once the image quality was optimized, the visual stimulation screen was clamped in position, and the experiment began.

Two-photon movies (512×512 pixels, 31 Hz for single plane and 512×512 pixels, 11 Hz for each plane in multi-plane experiments), eye tracking (30 Hz), and behavior (30 Hz) were recorded and continuously monitored. Recording sessions were ~ 1 hour long, but could be interrupted if any of the following was observed: 1) mouse stress as shown by excessive secretion around the eye, nose bulge, and/or abnormal posture; 2) excessive pixel saturation (> 1000 pixels) as reported in a continuously updated histogram; 3) loss of baseline intensity caused by bleaching and/or loss of immersion water in excess of 20%; 4) hardware failures causing a loss of data integrity.

Table 2. Maximum laser power used per depth range.

Depth (μm from surface)	Laser Power (mW)
0-50	0-30
50-100	25-50
100-150	50-80
150-200	70-100
200-250	90-125
250-300	110-170
300-350	150-180
350-400	160-190
400-450	200-240
450-500	200-240
500-550	200-240
550-600	200-240

A water immersion objective was used for imaging in single-plane mode on Scientifica microscope. It was supplemented at times while imaging using a micropipette taped to the objective (Microfil MF28G675 WPI) and connected to a 5 ml syringe via an extension tubing. This ensured that signal intensity deterioration did not occur due to the water evaporation. Intensity stayed within 5% of the original intensity level and allowed water level adjustments to be made without interrupting an experiment.

During multi-plane experiments, water-based ultrasonic gel was used as immersion medium. This allowed for better control over evaporation and water leakage while imaging on Multiscope.

At the end of each experimental session, a z-stack of images ($\pm 30 \mu\text{m}$ around imaging site, $0.75 \mu\text{m}$ step) was collected to evaluate cortical anatomy as well as z-motion during acquisition. In addition, a full-depth cortical z stack ($\sim 800 \mu\text{m}$ total depth, $5 \mu\text{m}$ step) was collected to document the imaging site location.

QUALITY CONTROL FOR 2-PHOTON CALCIUM IMAGING

Each experimental session was for data integrity based on a broad range of operational parameters. A comprehensive report was automatically generated to track data trends, animal behavior, experimental failures and errors. To minimize bias, rig operators did not review their own experiments, a secondary operator was assigned to review the QC report.

Assessment of the following quality metrics was performed after each imaging session. Failure to meet any of these criteria resulted in the session being retaken on a subsequent day.

1. Image Saturation: Initial movie frames were assessed to confirm that photonic saturation did not exceed 1000 pixels and the full dynamic range of the recording system was adequately covered.
2. Photobleaching: Epochs of fluorescence at the beginning and end of a session were compared to ensure that baseline fluorescence did not drop greater than 20%.
3. Field of View Targeting Validation: Targeted imaging locations were checked against the intrinsic signal imaging data, using a registered coordinate system, to confirm that data was collected from the correct visual area.
4. Z-Axis Stability: Stability of image recording was assessed by comparing a windowed average image from the first and last 5 minutes of the experiment to a z-stack of images ($\pm 30 \mu\text{m}$ around imaging site, $0.75 \mu\text{m}$ step) collected at the end of each experimental session to calculate the amount of drift that occurred over the session. Experiments with z-drift above $10 \mu\text{m}$ over the course of the entire session were excluded.
5. Animal Stress: Behavior videos were viewed to confirm that animals did not show excessive signs of stress. Any animal that showed eye secretion covering the pupil or excessive orbital tightening was returned to its home cage to recover. The presence of nose bulge, flailing and abnormal postures was also monitored.
6. Task Performance: Behavior performance in the change detection task was confirmed to have met (or exceed) a peak d-prime of 1.0.
7. Temporal Sync: Temporal alignment of data streams was confirmed.
8. Hardware/Software Failure: Multiple datastreams and metrics were assessed to ensure that incoming data integrity was not compromised by hardware and/or software related errors
9. Excessive Motion: Imaging frames were checked for residual motion (after motion correction algorithms had been applied)

10. Interictal Events: Presence of interictal events was assessed by measuring the full field fluorescence and calculating the intensity spike prominence and width for the first 10,000 frames of the 2P movie. Experiments with a non-zero probability of interictal events were then checked manually in order to exclude any potentially epileptic mice (see Steinmetz et al., 2017).

A final, container-level QC assessment was performed once all data collection for a mouse was completed. This secondary assessment included assessing the following metrics:

1. Full Container Status: Containers (the set of imaging sessions for a given field of view) were confirmed to contain all required datasets.
2. Brain Health: 2-photon serial tomography sections were examined to assess general brain health. Health assessment includes checking for excessive bruising, brain abnormalities, deformities, necrotic tissue damage, and checking for any signs of laser damage.
3. Cell Matching: Imaging fields of view were checked across all experiments in a container to confirm successful targeting of the same field of view. This assessment was made visually, based on similarity in patterns of vasculature and the presence of clearly identifiable matched cells across sessions.

SECTION F: DATA PROCESSING

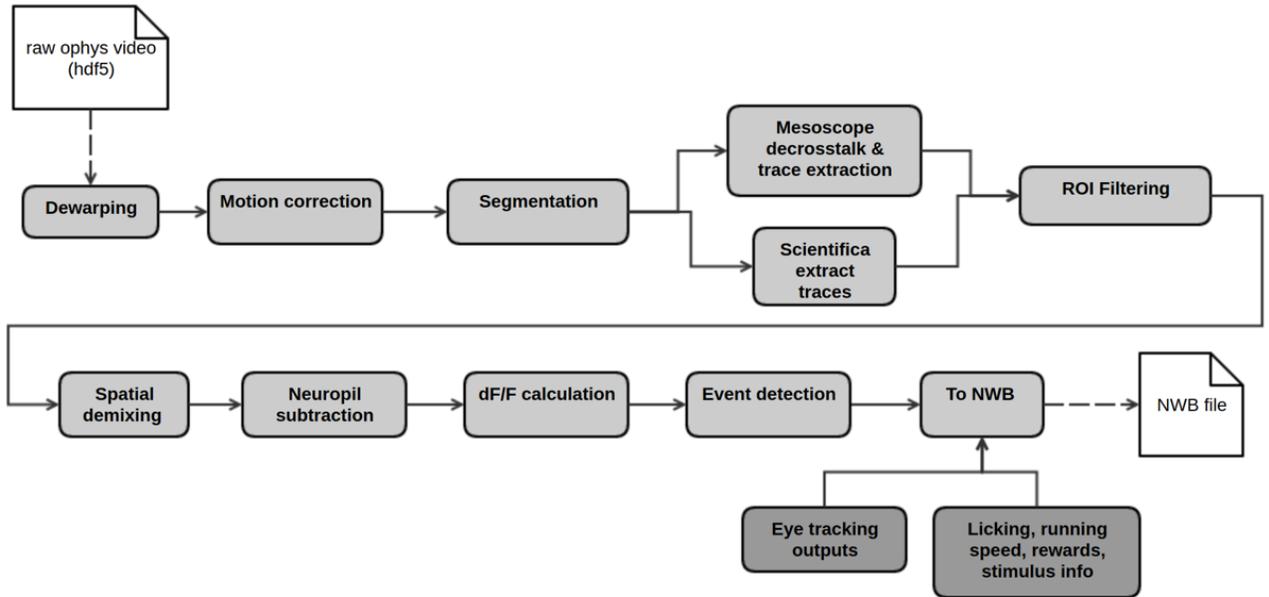


Figure 1: Overview of optical physiology data processing steps, starting with raw 2-photon movies and ending with packaging of processed data into NWB files

DEWARPING

To account for variation in scanning rate across each line in the 2p-imaging frame, due to the non-constant speed of the resonant scanner, data acquired with the Scientifica microscopes required a dewarping step (Multiscope instruments have built-in dewarping correction).

Correcting for the warping in the image involves taking a subset of the warped image's columns, towards the edges of the image where the scanning speed is lower, and combining them using parameters derived from calibration data acquired with a standard grid image. To determine correct dewarping parameters, the grid image is adjusted until the dewarped grid is uniform. Each side of the image is dewarped independently.

The following equations are used to determine which columns are chosen from the warped image and how they are combined. The j^{th} column of the dewarped image will be given by the formula:

$$col[j] = (1 + \lfloor g[j - 1] \rfloor - g[j - 1]) * input_image[\lfloor g[j - 1] \rfloor] + (g[j] - \lfloor g[j] \rfloor) * input_image[\lfloor g[j] \rfloor],$$

where

$$f(x) = x - \text{int} \left[b \left(1 - \sin \left(\left(\frac{x}{3a} + \frac{1.0}{6.0} \right) \pi \right) \right) + 0.5 \right], \quad x \in [0, a]$$

$$g(x) = (x + 0.5) - \left[b \left(1 - \sin \left(\left(\frac{x + 0.5}{3a} + \frac{1.0}{6.0} \right) \pi \right) \right) \right], \quad x \in [0, a]$$

Notice that this is taking a linear combination of the two (generally, adjacent) columns of the original image, using the decimal parts of the g function as coefficients. If these ever end up being non-adjacent columns from the original image, then we also add the value of the skipped column to $\text{col}[j]$.

In the event that both $f[j]$ and $g[j - 1]$ are not nonnegative, then we fill that column with the average from the original image. And if f is nonnegative but g is not, then we simply replace that column with the exact same column from the original image.

For each experiment, there are four parameters that tell us how much the image has been warped. These are called aL , aR , bL , and bR . The L and R refer to the side of the image which is being dewarped. In the equations above, a and b are replaced with their corresponding parameter, depending on whether you are dewarping the left side or the right side. The parameters aL and aR specify how far (in pixels) into the image the warping occurs. The parameters bL and bR are more a measure of how warped the image is in those areas.

MOTION CORRECTION

We used [Suite2P](#) v0.9.3 [rigid registration](#) for motion correction of 2-photon movies. Suite2P performs an iterative phase-correlation-based registration on a small subset of frames to generate a [reference image](#) from the average projection of those frames. The registration of the entire movie then proceeds with registration by phase correlation of each frame to this reference image. We saved Suite2P's registered output tiff stacks and concatenated them into a single hdf5 format compatible with the rest of our processing pipeline.

Suite2P's parameter *maxregshift* has a default value of 0.1, which clips lateral motions of more than 10% of the FOV dimension. We found a few examples in our release data where our experiments had real, long-timescale lateral shifts of greater than 10%. We increased the *maxregshift* parameter to 0.2, to allow for registration of these experiments. A consequence of this change was that up to 20% lateral shifts were allowed by Suite2P over short timescales as well. We observed that these short timescale shifts, often single frames, were likely not physical shifts, but struggles of the registration algorithm to register low signal-to-noise frames. Across an entire experiment, we monitor the worst-case shifts and establish a motion border which invalidates any ROI which touches it. To prevent too much inflation of the motion border exclusion area by these unphysical short timescale shifts, we detrended the x and y lateral corrections with a median filter over a 3 second window. We clip frame displacements to a +/-5% window around any outlier more than 5% above or below the detrended corrections was truncated to 5%. This allowed us to handle physical long timescale drifts and limit the impact of artifactual short timescale shifts. Any movie frames where the lateral shift was truncated were translated to the new truncated translation value.

For each movie, we produce the following outputs: the registered hdf5 movie for downstream processing, the motion correction shifts both from Suite2P directly and after our detrended clipping step, and Suite2P's output *corrXY*, the value of phase-correlation at the best alignment for each frame. For QC inspection, we generate plots showing the x and y lateral shifts and *corrXY* resulting from motion correction, as well as a video preview showing the movie before and after motion correction, side-by-side, averaged to 0.5 frames-per-second and a 10x playback speed.

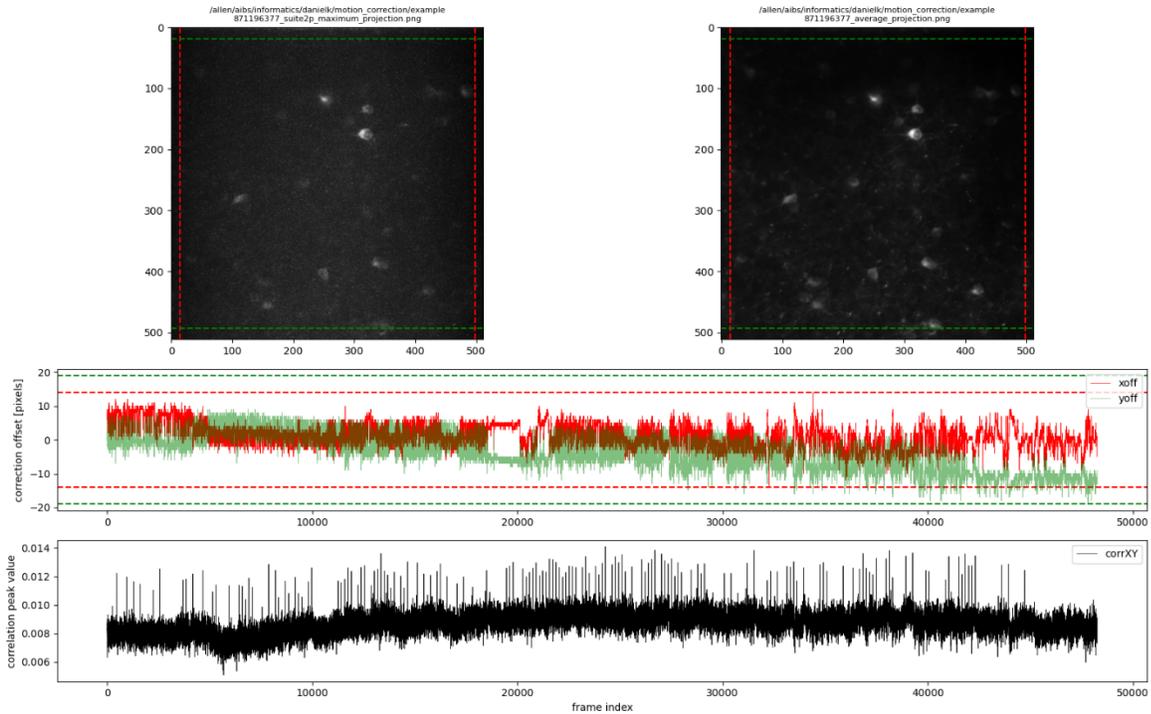


Figure 2. QC inspection plot of the motion correction processing step. Upper left panel: the maximum projection of the motion-corrected movie, with motion correction borders (exclusion zones) indicated in red and green dash lines. Upper right panel: the average projection of the motion-corrected movie, with motion correction borders (exclusion zones) indicated in red and green dashes. Upper horizontal panel: The determined x (red) and y (green) registration translations for each frame, plotted vs. frame index. Lower horizontal panel: corrXY, the value of phase-correlation of each frame with the inferred reference frame.

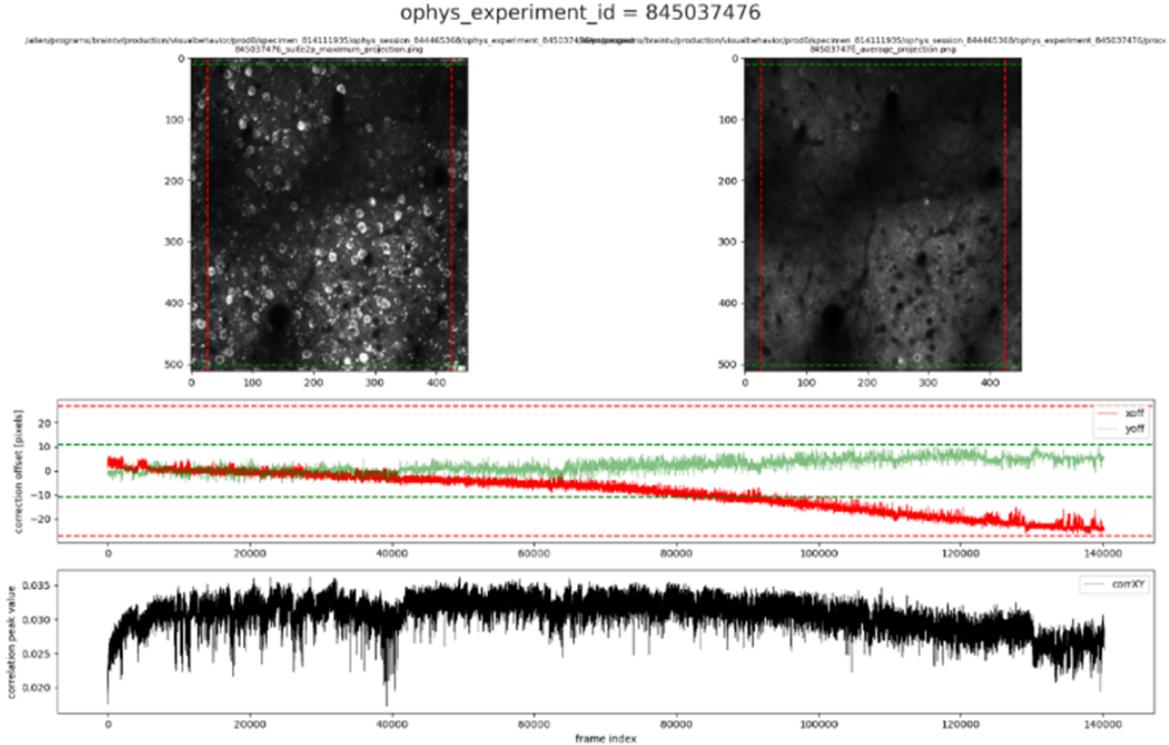


Figure 3. QC inspection plot of the motion correction processing step. Panel descriptions are the same as Figure 2. This experiment showed larger motion drift.

CELL SEGMENTATION

OVERVIEW

The Visual Behavior 2P project used the same segmentation procedure that was developed for the Visual Coding 2P dataset, published in de Vries et al., 2020. The active cell segmentation module was designed to locate active cells within a field-of-view (FOV) by isolating cellular objects using the spatial and temporal information from the entire movie. The goal of the active cell segmentation module is to achieve robust performance across experimental conditions with no or little adjustment, such as different mouse cell lines, fluorescent proteins (e.g., GCaMP6f or GCaMP6s), and FOV locations of visual areas and depths. The process begins with the full image sequence as input to apply both the spatial as well as temporal information to isolate an individual active cell of interest without data reduction, such as by PCA, and does not make assumptions about the number of independent components existing in the active cell movie. Also, in contrast to other methods, this approach separates the individual steps, including identifying and isolating each cellular object, computing confidence of each identified object (by object classification) and the step of resolving objects overlapping in x-y space (which lead to cross talk in traces), so that each can be improved upon if necessary.

PRESEGMENTATION.

The motion corrected image sequence was spatially median filtered (using 3x3 pixel kernel) to reduce white noise. The sequence was then low pass filtered and downsampled by 1/8 temporally to enhance the signal-to-noise ratio (SNR). The processed image sequence was then divided into periods of fixed temporal length p , where $p = 50$ frames (~13.3 sec.). The maximum projection image from each period and the mean image (μ_image) of the whole sequence were computed. The maximum projection image from all temporal periods, called Periodical Projection frames $PP(t)$, were further normalized to become Normalized Periodical Projection (NPP) frames:

$$NPP(t) = MF((PP(t) - \mu_image), 3x3) * G(t)$$

Where $G(t)$ is the frame intensity normalization gain computed based on the intensity histogram of each $PP(t)$. This is to normalize any change in overall intensity across the experiment, and to reduce experiment-to-experiment variability. $MF(3x3)$ is median filtering with a 3x3 pixel kernel.

Note in each $NPP(t)$, a subset of cells can be found with changes in fluorescence during that time period. With sufficient experiment length, and with many sweeps of different stimuli, various repetitive cell firing patterns can be found. Cells with overlapping spatial positions in x and y can be observed as firing at different time frames, allowing the following detection process to identify them individually despite having spatial overlap.

DETECTION

Adaptive and mathematic morphological image processing techniques were applied to process each $NPP(t)$. After band-pass filtering, an initial binary object map was generated by thresholding the resulting image minus a low pass version of itself to capture spatially varying background intensity. Conditional dilation, erosion and connected component analysis were applied to filter the candidate binary objects and fill holes. The final set of regions of interest (ROIs) in each NPP were identified using another connected component labelling and a simple rule-based classifier. This classification was based on comparing measured morphometric attributes (object area, shape, intensity, uniformity, etc.) of the objects to the statistics derived from the targeted active cell components in the sample data sets. After each frame was processed, a set of candidate ROIs from each NPP were then grouped with candidate ROIs from all other NPPs. ROIs within a similar spatial location (defined by the distance between centroids $< 5 \mu\text{m}$) and with similar morphometric attributes (e.g., delta (area or shape) $< 20\%$) across NPP frames were grouped as the same cell object. The ROI with the highest contrast and with shape and area within range statistically derived from sample data was selected to represent that cell in the mask image and in a composite image for visual QC. ROIs with different spatial locations (centroids $> 5 \mu\text{m}$ apart) and or dissimilar morphometric attributes are recorded as different cells.

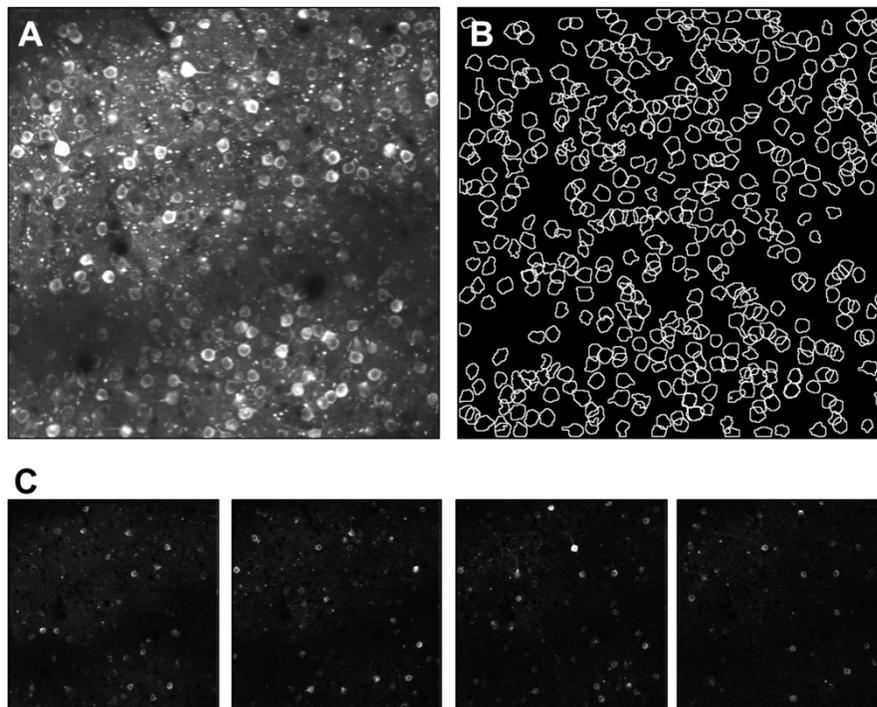


Figure 4. Pre-segmentation. **A**, Maximum projection image of an experiment. Note some cell bodies are not clearly visible in the busy background and neuropil, often due to overlap with other objects. **B**, Boundaries of detected active cells by the segmentation algorithm is shown. Many overlapping cellular objects not easily visible in the projection image were successfully identified and isolated because detection operates in both temporal and spatial space. **C**, Examples of consecutive NPP(t) images. Contrast of firing cells are enhanced by reducing background (see definition of NPP(t)).

Occasionally two or more spatially overlapping ROIs could be found that were active in the same timeframe and were therefore detected as a single ROI for that NPP. Additional steps to classify them as multiple-cell objects were taken using their attributes of combined area and shape (eccentricity).

After ROI detection in each NPP frame and grouping of all ROIs across frames was completed, a set of “unique” active cell objects was identified (Figure 4B). Cells near the FOV boundaries ($3\ \mu\text{m}$) were eliminated from further consideration due to the fact that motion shifts can create boundary effects to the traces computed from these cells. To generate the segmentation mask image, all non-overlapping cells were placed in a single mask plane and overlapping cells were placed in subsequent planes, to ensure unique identification of all cells (Figure 4C).

CROSSTALK REMOVAL IN MULTISCOPE DATA

Crosstalk between focal planes is a fundamental limitation in multiplexed microscopy systems, such as the Multiscope. Details about system hardware and methods used to optimize crosstalk in the data are described above and can be found in Lecoq et al., 2020. Crosstalk removal was performed on fluorescence traces using an ICA-based approach (implementation `scikit-learn.FastICA`), where independent components are estimated by minimizing Gaussianity of the data (Hyvärinen, 1999). The assumption is that the two planes are a mixed observation of two clean sources and are mixed linearly using a mixing matrix. We assume a mixing matrix of the form $\begin{bmatrix} 1-a & a \\ b & 1-b \end{bmatrix}$, where a and b are in $[0,1)$, and probably around 0.15. After FastICA, we transform the resulting mixing matrix to be of this form to recover the proper scaling of the mixed signals. Prior to FastICA, data undergoes whitening; we do not use the built-in whitening of the FastICA module (as it appears to contain a bug that affects scaling of the outputs).

Algorithm description

The plane whose traces we are correcting is referred to as the **signal plane**. The plane coupled to it (simultaneously acquired with temporal multiplexing) is referred to as the **crosstalk plane**. The algorithm is run on both permutations of signal plane and crosstalk plane, i.e., for each pair, it is run once with plane A as the

signal plane and plane B as the crosstalk plane and once with plane B as the signal plane and plane A as the crosstalk plane.

Cell segmentation is performed on each plane independently to generate cell ROIs. We define the set of ROIs as those detected in the signal plane. We construct **raw signal traces** by measuring the average per-pixel flux in each ROI in the motion-corrected movie taken from the signal plane (trace extraction). We construct **raw crosstalk traces** by measuring the average per-pixel flux in the same ROIs in the motion-corrected movie taken from the crosstalk plane (these ROIs are those detected in the signal plane). For both the raw signal traces and the raw crosstalk traces, raw neuropil traces are constructed using the pixels bordering the ROIs. We flag any ROI whose footprint or neuropil intrudes on the motion correction border as invalid and remove it from further processing.

Signal and crosstalk planes are coupled and leak into to each other due the imperfections of the temporal multiplexing approach. We assume that, for each ROI, the measured, or raw signal and crosstalk traces are linear combinations of clean, or unmixed signal and crosstalk traces, i.e.

$$R = M U$$

where R is a $2 \times N$ matrix (N is the number of timesteps in the traces) such that $R[0,:]$ is the raw signal trace and $R[1,:]$ is the raw crosstalk trace. U is a $2 \times N$ matrix such that $U[0,:]$ is the unmixed signal trace and $U[1,:]$ is the unmixed crosstalk trace. M is the 2×2 mixing matrix relating the two. For each ROI, $U[0,:]$ represents the trace coming from that ROI uncontaminated by crosstalk. That unmixed signal is the trace that we want to use in all future processing steps. We use Independent Component Analysis (ICA) [1] to solve for U and M . Specifically, we:

- 1) Subtract the mean from each trace and find the 2×2 matrix W which will transform (R -mean) into a “whitened” dataset whose correlation matrix is approximately the 2×2 identity matrix.
- 2) Use [scikit learn's FastICA](#) algorithm to solve for U and M
- 3) The scale of the traces is corrected by computing the scaling that will restore the mixing matrix to the assumed form of $[[1-\alpha, \alpha], [\beta, 1-\beta]]$. This scaling can be computed by multiplying the inverse of the mixing matrix by the vector $[1, 1]$. This transformation is applied to the traces to restore the original scale.
- 4) If the off-diagonal elements of M are positive and less than 0.3, ICA has converged to a valid result. Identify the unmixed trace that is most closely correlated with the raw signal trace as the unmixed signal trace (ICA is agnostic regarding the ordering of unmixed signals).
- 5) If ICA failed to converge to a valid result, the ROI is marked as having failed ICA.

We use all the ROIs that pass the above algorithm to construct an average mixing matrix M for the signal plane. The inverse of this matrix is used to unmix any ROIs marked as “failed” in step (4). We find unmixed neuropil traces by taking the mixing matrix used for the associated ROI and multiplying its inverse by the matrix of raw neuropil traces. If no ROIs in the plane converged to a valid result, it is impossible to construct an average mixing matrix for the plane and we mark the plane as having failed processing.

We run the unmixed signal and unmixed crosstalk traces through an event detection algorithm described in Giovannucci et al. 2019. We define events as a sequence of five timestamps whose sum of log probabilities exceeds a threshold (14 in our implementation). We calculate the probability relative to a Gaussian distribution centered on the mode trace value whose width is calculated considering the distribution of all trace values less than the mode. We define each timestamp that passes this test as an event in the trace.

For each ROI, we assess whether any events occur in the unmixed signal trace that are independent from the events in the unmixed crosstalk trace. We define independent events as those events in the unmixed signal trace which do not occur within two timestamps of an event in the unmixed crosstalk trace. If no such independent events occur in the unmixed signal trace, we mark the ROI as a ghost (Figure 5, A) (i.e. an ROI

that was only detected because of crosstalk contamination from the coupled plane) and discard it from future processing steps.

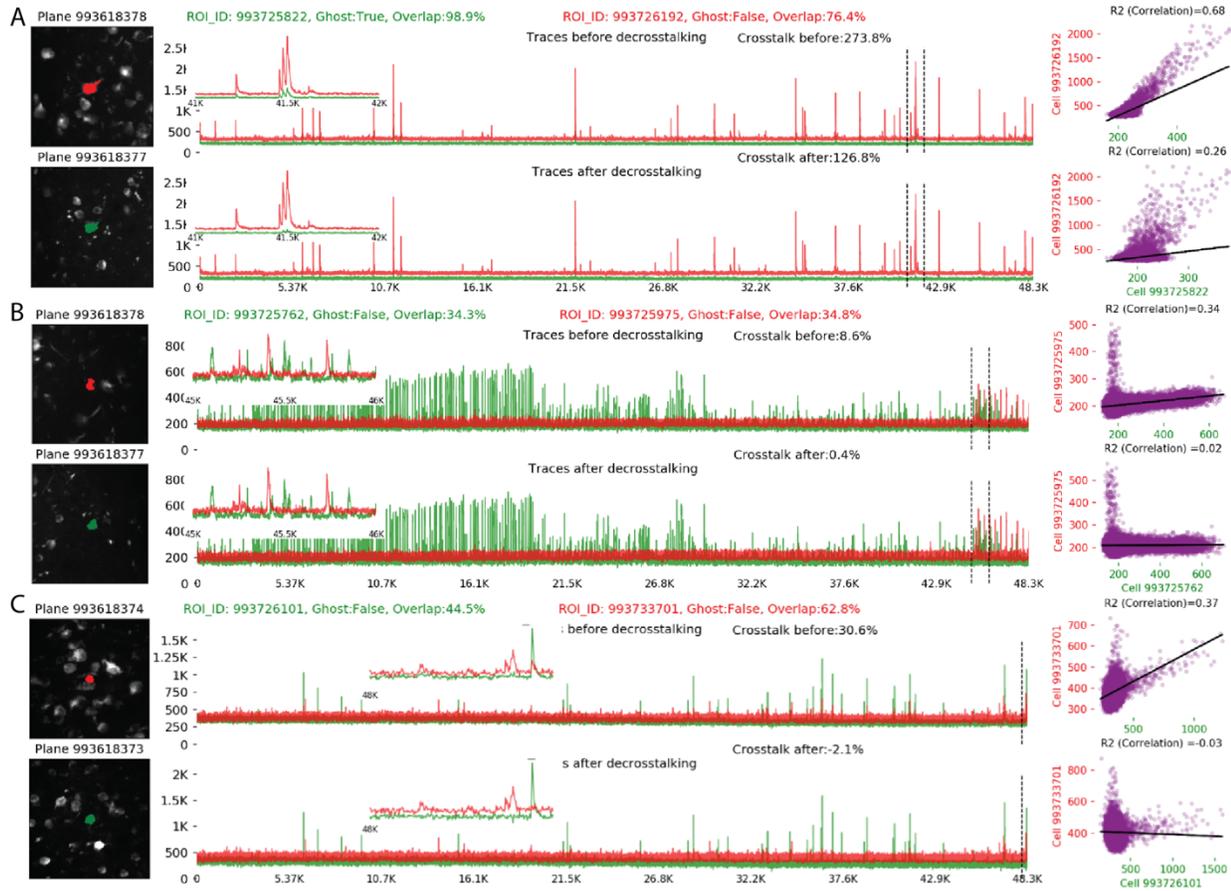


Figure 5. Examples of unmixed traces. A. Ghost cell (green) has no independent signal in the crosstalk plane compared with signal plane. The algorithm detects this cell as a Ghost cell and excludes from the dataset. **B.** An example with minimal crosstalk between the two planes. **C.** Two overlapping neurons where each has independent activity compared to the other, but correlated activity in red ROI is removed by the decrosstalk algorithm

ROI FILTERING

Not all ROIs generated by segmentation are complete individual cell bodies. To exclude ROIs that are not actually cell bodies from further analysis, the ROIs are labeled with a multi-label classifier where any ROIs that end up labeled are not considered cell bodies. The set of reasons to exclude an ROI are: the ROI is a union of two or more cells; the ROI is a duplicate of another; the ROI is close to the edge of the FOV and is impacted by motion such that parts of the ROI are missing from the video; the ROI is likely an apical dendrite and not a cell body; or that the ROI is too small, too narrow, or too dim to confidently be considered a cell body.

The initial ROI filtering was generated by a set of heuristics based on depth, shape, area, intensity, signal-to-noise, and the ratio of mean to max intensity of the max projection. The initial filtering was used to generate a set of training labels, on which a multi-label classifier was trained. The multi-label classifier is implemented using a linear Support Vector Classifier trainer for each label (binary relevance) using metrics generated in segmentation combined with depth, driver, reporter, and targeted structure as features. The final ROI filtering workflow is to 1) label ROIs that fall within the motion cutoff regions at the border, 2) label ROIs using the binary relevance classifier, 3) label significantly overlapping ROIs as duplicates, and 4) label ROIs that significantly overlap two or more ROIs as unions.

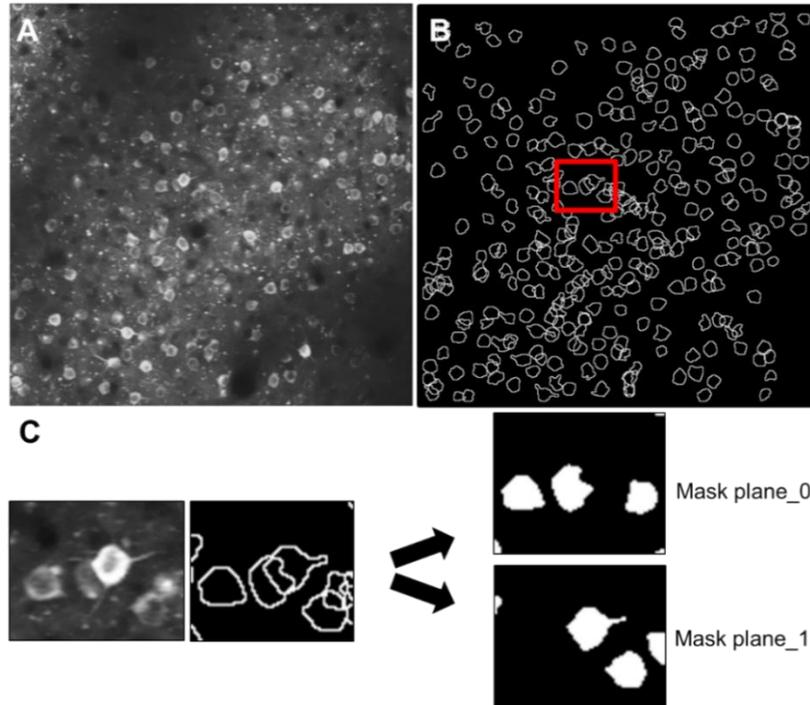


Figure 6. ROI detection. **A**, Maximum projection image of another experiment. **B**, Boundaries of detected active cells by the segmentation algorithm is shown. One area with overlapping cells is highlighted in red. **C**, Mask of detected overlapping cells are placed in different planes for trace calculation.

DEMIXING TRACES FROM OVERLAPPING ROIS

The simplest way to extract fluorescence traces, given a set of ROI masks, is to average the fluorescence within each ROI. If two ROIs overlap, this procedure will artificially correlate their traces. Therefore, a model is used where every ROI has a trace which is distributed across its ROI in some spatially heterogeneous, time-dependent fashion:

$$F_{it} = \sum_k W_{kit} T_{kt}$$

where W is a tensor containing time-dependent weighted masks: W_{kit} measures how much of neuron k 's fluorescence is contained in pixel i at time t . T_{kt} is the fluorescence trace of neuron k at time t - this is the desired value to estimate. F_{it} is the recorded fluorescence in pixel i at time t .

Importantly, this model applies to all ROIs, including those too small to be a neuron or otherwise filtered out. Duplicate ROIs (defined as two ROIs with >70% overlap) and ROIs that are the union of two other ROIs (any ROI where the union of any other two ROIs accounts for 70% of its area) are filtered out before demixing, and the remaining filtering criteria are applied after demixing. Projecting the movie (F) onto the binary masks (A) reduces the dimensionality of the problem from 512x512 pixels to the number of ROIs:

$$\sum_i A_{ki} F_{it} = \sum_{k,i} A_{ki} W_{kit} T_{kt}$$

where A_{ki} is one if pixel i is in ROI k and zero otherwise—these are the ROI masks from segmentation, after filtering out duplicate and union ROIs. At a particular time point t , this yields the simple linear regression:

$$AF(t) = (AW^T(t))T(t)$$

where the weighted masks W are estimated by the projection of the recorded fluorescence F onto the binary ROI masks A . On every imaging frame t , the linear least squares solution T are computed in to extract each ROI's trace at that time point.

It was possible for ROIs to have negative or zero demixed traces T . This occurred if there were union ROIs (one ROI composed of two neurons) or duplicate ROIs (two ROIs in the same location with approximately the same shape) that the initial detection missed. If this occurred, those ROIs and any that overlapped with them were removed from the experiment. This led to the loss of $\sim 1\%$ of ROIs.

NEUROPIIL SUBTRACTION

The recorded fluorescence from an ROI was contaminated by the fluorescence of the neuropil immediately above and below the cell due to the point-spread function of the microscope. In order to correct for this contamination, the amount of contamination was estimated for each ROI. The estimated F_N was done by taking an annulus of $10\ \mu\text{m}$ around the cellular ROI, excluding pixels from any other ROIs. In order to remove this contamination, the extent to which ROI was affected by its local neuropil signal was evaluated.

The recorded traces were modeled as F_M as $F_M = F_C + rF_N$, where F_C is the unknown true ROI fluorescence trace and F_N is the fluorescence of the surrounding neuropil. In order to estimate the contamination ratio r for each ROI, the error was minimized $E = \langle (F_C - (F_M - rF_N))^2 + \lambda \Lambda(F_C) \rangle_t$ by jointly optimizing for r and F_C . $\Lambda(F_C)$ is the first temporal derivate of the cellular trace, weighted by $\lambda = 0.05$; this smoothness constraint on the cellular trace allows per-ROI optimization for r . $\langle \cdot \rangle_t$ denotes an average over time. Gradient descent was used on r . At each step of the gradient descent, F_C was solved at the zero gradient of E . Gradient descent was performed on the first half of the traces and computed E on the second, so that it is a cross-validation error. After computing r and F_C for an ROI, the neuropil-subtracted trace $F_M - rF_N$ was used as the basis for all subsequent analysis in order to avoid any residual effects of the smoothness constraint.

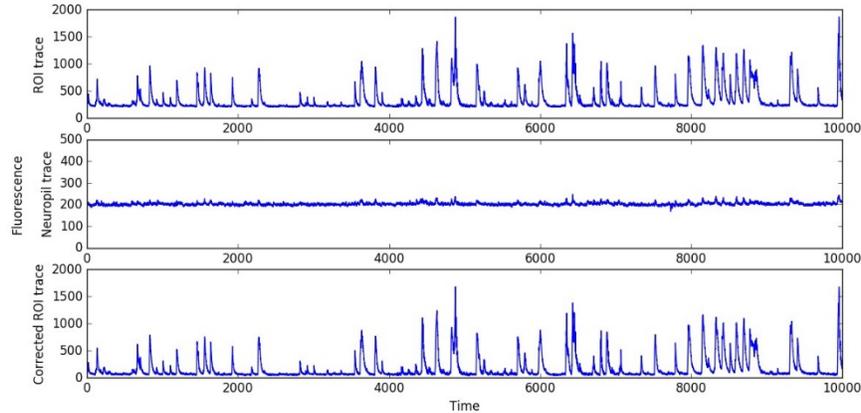


Figure 7. Example of original ROI and neuropil traces, and neuropil-subtracted ROI trace.

ROI 149 from experiment 502199136, with $r=.79$.

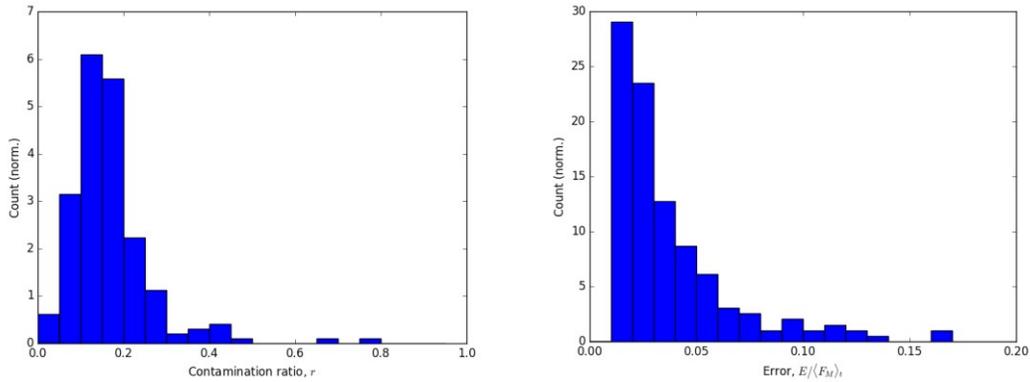


Figure 8. Distribution of contamination ratios (left) and cross-validation errors (right) for an example experiment (502199136). For this experiment, neuropil subtraction failed on 8% of cells (16/199).

To standardize the learning rate and initial conditions of the gradient descent, each ROI’s neuropil trace was normalized to (0,1). The measured ROI trace was normalized by the same amount, used a learning rate of 10 and initial condition of $r = 0.001$. The gradient descent was stopped at the first local minimum of E . If the resulting r was greater than 1 or less than 0, or final cross-validation error E greater than $2 | \langle F_M \rangle_i |$, the gradient descent was attempted again with a 10x slower learning rate. If those convergence criteria were still not met, an initial condition of $r = 0.5$ was used. If those convergence criteria were still not met, that ROI was flagged and after computing r for all other ROIs in the experiment, set r for un-converged ROIs to the mean.

To validate the performance of our algorithm, it was tested on a publicly-available benchmark dataset (Chen et al., 2013). A distribution of contamination ratios was obtained, centered nearly on the author’s choice of 0.7 (mean r of 0.68 vs their choice of 0.7), but with significant heterogeneity (Figures 7, 8). For this benchmark dataset and using the same optimization parameters as for the **Allen Brain Observatory – Visual Coding** data, 6 of 36 cells failed the initial neuropil subtraction with $r > 1$, and would have gone through the additional steps outlined above.

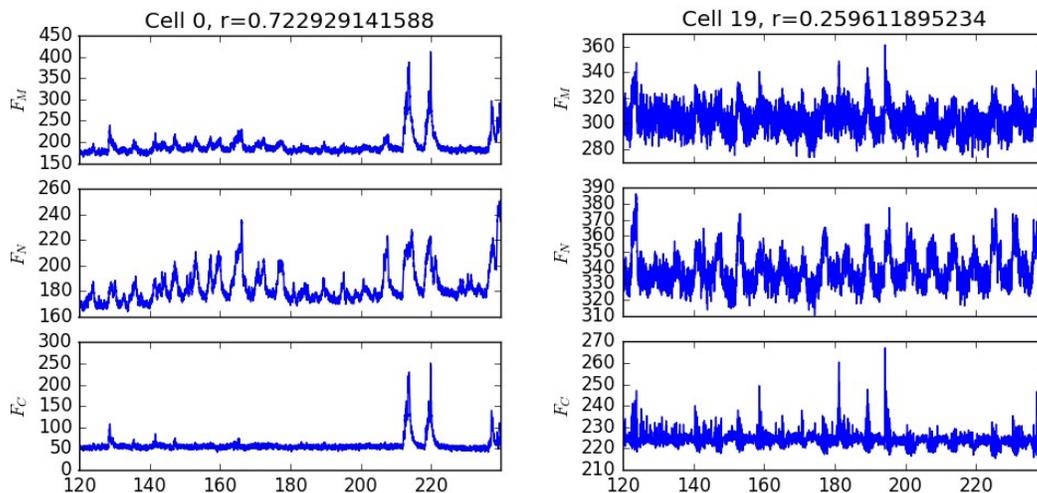


Figure 9. Example results of neuropil subtraction from the benchmark dataset. Fluorescence vs time (sec).

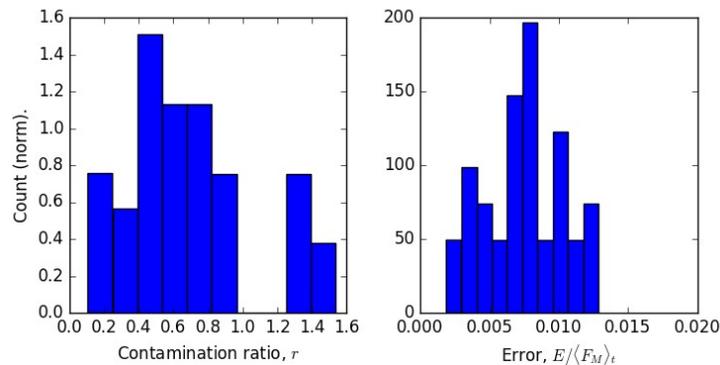


Figure 10. Summary of neuropil subtraction on a benchmark dataset. Left, histogram of contamination ratios on the 36 cells of the GCaMP6f data of Chen et al., 2013. Mean: 0.68, Standard deviation: 0.38. Right, histogram of the corresponding cross-validation errors.

DF/F CALCULATION

We return normalized, detrended traces of neuronal activity by performing the following algorithm on each trace. First, we estimate the standard deviation of the noise in the trace. We do this by centering the trace on the curve resulting from applying a median filter with kernel size 3.33 seconds to the trace. To avoid including signal events in our estimate of noise, we discard any values that exceed 1.5 times the absolute value of the minimum of this centered trace. We make a first estimate of the standard deviation of noise in the trace as 1.4826 times the median absolute deviation of this truncated, centered trace. We further discard any values that exceed 2.5 times this first estimate of the standard deviation and finally return 1.4826 times the median absolute deviation of the remaining centered trace values as the standard deviation of noise in the trace.

Next, we calculate the dF/F trace. We define the baseline activity in the trace as the result of applying a median filter with kernel size 600s to the trace. We subtract this baseline from the raw trace and normalize the difference by the baseline. At timesteps where the baseline trace is less than the standard deviation of noise calculated above, we normalize by the standard deviation of noise, instead. This gives the dF/F trace.

Finally, we detrend the dF/F trace. We estimate the trend in the dF/F trace by applying a median filter with kernel length 3.33 seconds. To prevent anomalously large trends from arising, we use the same algorithm we used to estimate the standard deviation of noise in the raw trace to estimate the standard deviation of noise in the dF/F trace. We constrain the trend to never exceed 2.5 times this estimated standard deviation of noise in the dF/F trace. We subtract the trend from the dF/F trace and report this as the detrended dF/F trace.

SESSION TO SESSION CELL MATCHING

Multiple 2-photon calcium imaging movies were acquired for each imaging plane across multiple imaging sessions. To map cells between sessions, we used an automated matching algorithm. The module has 4 steps:

1. Determining the spatial transform between each pair of sessions using image alignment techniques on the average projection images of the sessions.
2. Applying the derived spatial transform to segmented cell masks.
3. For each pair of sessions, solving the linear assignment problem (i.e. bipartite graph matching) to determine which pairs of masks are most likely to be related.
4. A graph combination method was used to join all the bipartite graphs to determine the most likely label sets that match across all sessions.

Determining the spatial transformation

The module first used an intensity-based method to register the average intensity projection images of each pair of sessions, producing a Euclidean transformation that registered each image pair. We found that a single image registration strategy did not work for all cases. In some cases, the true lateral shifts between two experiments were too large for the ECC registration algorithm. In other cases, the border in the projection images biased the

registration towards matching borders and not cell soma. In still other experiments, a histogram equalization step that helped most experiments resulted in a degraded registration.

We implemented a “meta-registration” where 4 different registration sequences were attempted. Each sequence employed an initial cropping step to eliminate the motion border problem. The sequences employed varying combinations of ECC, PhaseCorrelation, and contrast adjustment. After each of the 4 sequences were attempted, we chose the best candidate based on the structural similarity metric (SSIM) between the two registered images. If none of the candidate sequences improved SSIM beyond that achieved with unregistered images, we flagged the registration as a failure for QC inspection.

In a few cases, there were still some unexplained failures to register. In these cases, we found that the average projection intensity images had low contrast, but the maximum projection intensity images had more visible features. By substituting in the maximum projections for the registration step, we achieved good registration and good cell matching for these containers.

Applying the spatial transformation to cell masks

The cell masks were materialized into a set of images, such that no ROIs overlapped in any one image. We then applied the relative transformation found from the intensity projections to these materialized images, to get the cell masks in an aligned space.

Solving the linear assignment problem between 2 sessions

To map cells, a bipartite graph matching algorithm (the Blossom method) was used to find correspondence of cells between sessions. The algorithm used cell labels in the pair-wise experiments as nodes, and the edges of the graph were weighted with weights, w :

$$w = \frac{d}{d_{max}} + \left(1.0 - \frac{I}{U}\right)$$

where d is the distance between the 2 mask centroids (in pixels), $d_{max} = 10$ pixels, I is the number of pixels that are shared between the 2 masks (the intersection), and U is the total number of pixels covered by the two masks combined (the union). The graph edge for any pair of masks with $d > 10$ pixels was assigned a very large weight, eliminating that pair as a candidate match.

The original version of this module used the “Hungarian” method to minimize the total weight. SciPy has an [equivalent implementation](#) of that method. We found that the solution from the “Blossom” method was more robust to order permutations of the inputs. We used the [networkx implementation](#) of the “Blossom” method. As this implementation seeks to maximize total weight, we inverted the weights, taking care to prevent division-by-zero:

$$w = \frac{1}{1 + w}$$

By maximizing the summed weights of edges, the bipartite matching algorithm found the “best match” between cells in each pair of experiments.

Joining the pairwise bipartite graphs

All the pairwise bipartite graphs were naively combined. This combined graph is not constrained to avoid labeling conflicts. For example, consider the simple case shown in the diagram below.

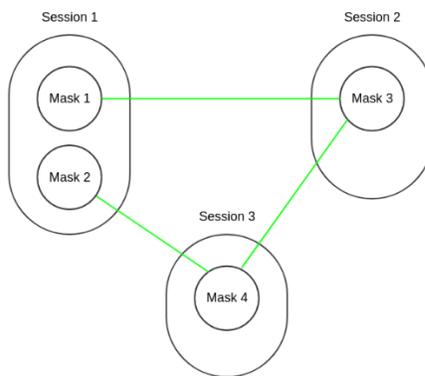


Figure 11. Cell Matching example illustration

The green lines indicate matches found from solving the pairwise linear assignment problem between sessions. In this naively combined graph, Mask 1 and Mask 2 are identified as being the same cell. Because they are distinctly identified masks within the same session, this is not possible. We resolve these labeling conflicts by iteratively removing the highest weight edges from sub-graphs until conflicts are resolved.

Matching was performed on the full set of ROIs identified by the segmentation algorithm, but ROIs that were deemed invalid by the ROI filtering step were not included in the final set of matched cells.

For quality control checks, we produced standard plots of pairwise registrations across sessions (Figure 12), as well as quantification of the fraction of ROIs matched across sessions (Figure 13) Session pairs with visible failures to register (clear features in the image were not aligned), or with very low fraction of matched cells (<10% matching between session pairs), were manually evaluated and an attempt was made to optimize the registration process to produce accurate registration. Session pairs that were not possible to register were excluded from the dataset.

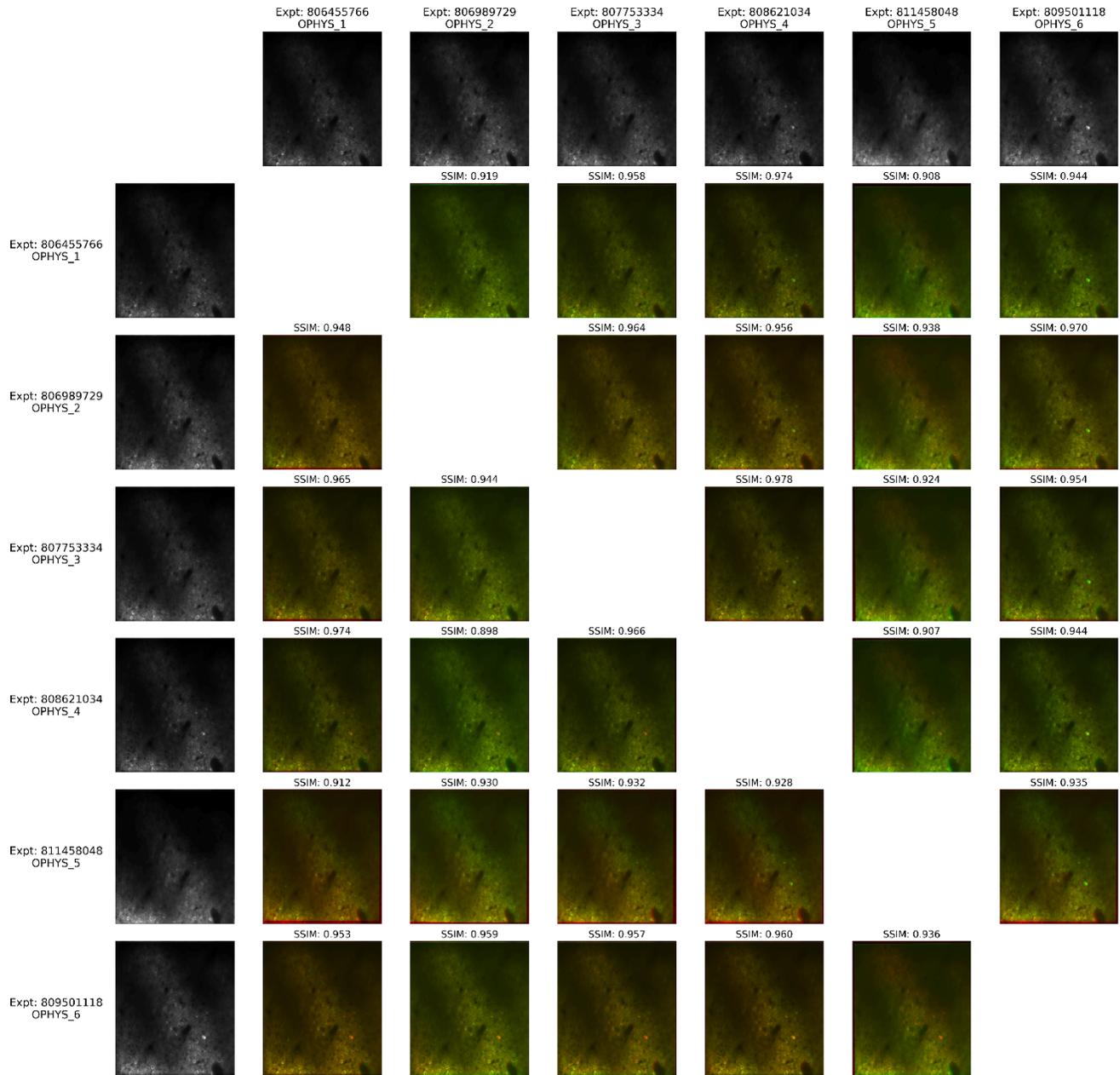


Figure 12. QC inspection artifact for the cell matching registration step. The average projection images, after spatial transformation, for all the sessions are shown in gray along the left column along the top row. Alignment of the two sessions' registered average images are shown as an RGB overlay, with one session in the red channel and the other in the green channel. Registration failures can be identified when the red and green images clearly do not overlap. A metric for [structural image similarity](#) (SSIM) is also included to quantify the degree of overlap (similarity) in the two images.

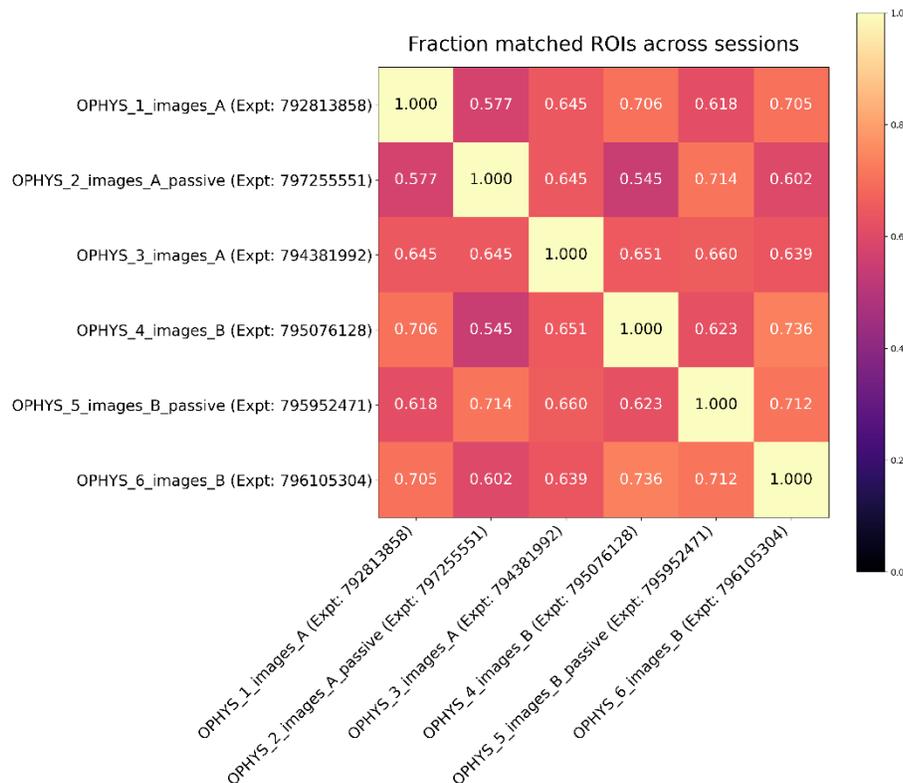


Figure 13. QC inspection artifact for pairwise matching. The color scale indicates the fraction of all possible ROIs that were matched in a pairwise assignment solution. When a particular row or column of this plot is much different from the rest of the table (for instance, fractions = 0), it provides a quick indication that there may be a data or processing problem to investigate.

The code for performing field of view registration and cell matching is tested in a CI/CD system and [open-sourced](#).

EVENT DETECTION

We used the [FastZeroSpikeInference](#) (“FastZero”) to identify events in traces derived from the 2P movies. FastZero fits a trace to a sum of exponentially decaying spikes, while imposing an L0 regularization, trying to minimize the number of events used to fit. The relative weight in the optimization penalty between number of events used and best residuals is controlled by an unspecified regularization factor. We iteratively searched the regularization factor space to find the regularization factor where the smallest event provided by FastZero was some multiplicative factor times the estimated noise level of the trace.

The multiplicative factor for experiments recorded at 31Hz (Scientifica rigs) was determined by testing an array of factors against ground truth data manually annotated by a human expert, and set to 2.0. Multiscope experiments were recorded at 11Hz. Use of the same multiplicative factor resulted in many more small amplitude events than in the 31Hz data when probed with synthetic calcium data. One can rationalize this difference intuitively by noting that the characteristic timescales of fluorescence decay do not change based on sampling rate, but the noise estimation does. To account for the effects of sampling rate on the noise estimation, for Multiscope, we empirically determined the equivalent multiplicative factor to be 2.6 by identifying the value that minimized the discrepancy between event magnitude traces extracted (using factor of 2.0) from Scientifica data sampled at 31 Hz, and event magnitude traces extracted from the same data but after downsampling by a factor of 3.

EYE TRACKING DATA PROCESING

A standardized pipeline was built for fitting ellipses to the pupil, eye (visible perimeter of the eyeball), and corneal reflection of the right eye, based on points tracked using the open source software DeepLabCut (CITE). DeepLabCut, which uses a pre-trained ResNet 50 deep residual network, was used to track (up to) 12 points along the perimeters of the eye, pupil, and corneal reflection. Ellipses were then fit to the tracking points and the ellipse fit parameters were saved to disk. Validation against hand-annotated 'ground truth' frames confirmed that a single 'universal' model, trained on a broad selection of data samples, robustly generalized on held-out data across different physiology rigs and individual animals.

All code for processing of the eye tracking data is visible in the [brain_observatory.behavior.eye_tracking_processing](#) module of the AllenSDK. The eye tracking data is available as the 'eye_tracking' dataframe in the session object. The dataframe has the following columns:

- Timestamps: the timestamp of every frame
- {pupil, eye or cr}_center_x: the x-position of the center of the pupil, eye or corneal reflection ellipse fit on that frame, in pixel space of the eye tracking movie frame.
- {pupil, eye or cr}_center_y: the y-position of the center of the pupil, eye or corneal reflection ellipse fit on that frame, in pixel space of the eye tracking movie frame.
- {pupil, eye or cr}_width: the (more) horizontal halfaxis of the ellipse fit of the pupil, eye or corneal reflection
- {pupil, eye or cr}_height: the (more) vertical halfaxis of the ellipse fit of the pupil, eye or corneal reflection
- {pupil, eye or cr}_phi: the angle to the (more)horizontal axis of the ellipse fit of the pupil, eye or corneal reflection, measured CCW from the x-axis of the image frame.
- likely_blink: a Boolean defining frames that have been identified as likely outlier fits, which is often caused by blinking/squinting of the eye.
- pupil_area: the area of the pupil, assuming that the pupil is a circle with a diameter defined by the major axis of the ellipse fit. Set to NaN where likely_blink == True.
- pupil_area_raw: the area of the pupil, assuming that the pupil is a circle with a diameter defined by the major axis of the ellipse fit. No outliers/likely blinks are removed.
- {eye or cr}_area: the area of the eye or corneal reflection ellipse fit. Set to NaN where likely_blink == True.
- {eye or cr}_area_raw: the area of the eye or corneal reflection ellipse fit. No outliers/likely blinks are removed.

Briefly, the ellipse fit parameters produced by the processing pipeline were loaded for each session. It was assumed that the pupil is actually round, but when viewed obliquely, it appears as an ellipse, the major axis of which reflects the pupil diameter. Thus, the area of the pupil is calculated on every frame as the area of a circle with a diameter defined by the longest axis of the ellipse on that frame. The area of the ellipses for both the corneal reflection and eye ellipse fits are calculated using the standard formula for the area of an ellipse.

On frames where the animal is blinking, tracked points may be missing or the confidence of tracked points can be low, and ellipse fits either fail completely or may be fit erroneously. To avoid including these erroneous fits in analysis, an algorithm attempts to identify these blink frames, adding a column to the eye_tracking dataframe called 'likely_blink'. Likely blinks are identified as frames where either the eye or pupil fit is missing, or where the z-scored value of the eye or pupil areas exceeds 3. In addition, two frames before and after every likely_blink identified by the above methods are also labeled as likely blinks. This is to avoid the possibility of analyzing erroneous fits caused by a partially opened eye. Below is an example of a series of frames surrounding an identified blink:

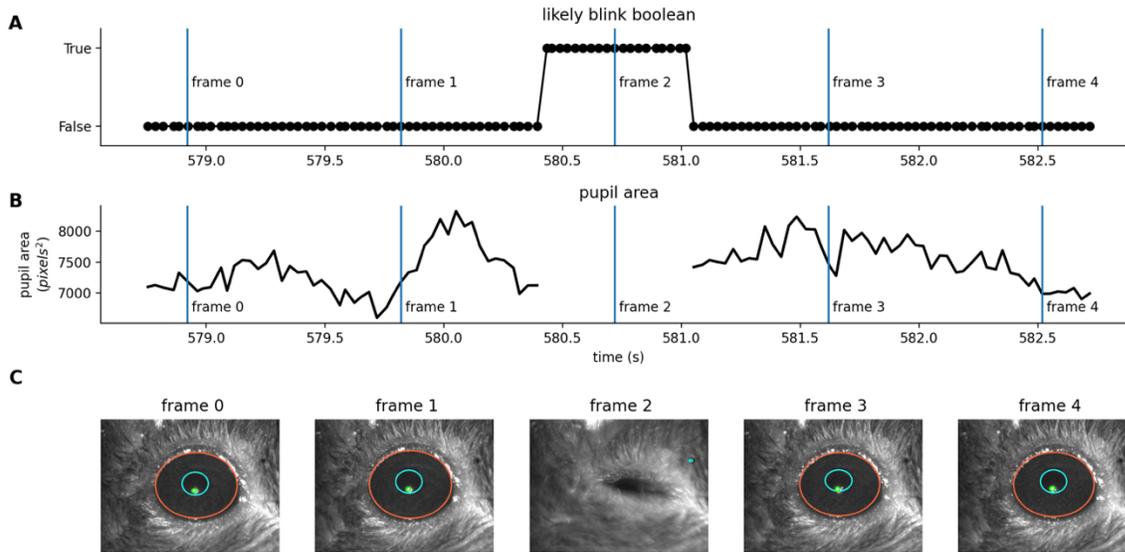


Figure 14. Eye tracking values surrounding an identified blink. A) The state of the 'likely_blink' Boolean for every frame. B) The calculated area for the pupil fit on every frame. Note that the area is filled with NaN where likely_blink == True. C. Five sample frames with eye and pupil fits overlaid, for the 5 frames annotated with vertical lines in A and B. Note that the pupil fit (in cyan) is grossly incorrect when the animal's eye is closed.

It is important to note that the outlier frames identified by the likely_blink algorithm are not always caused by blinks. Instead, they may be frames where the DeepLabCut algorithm simply failed to identify a reasonable fit, triggering the outlier detection portion of the algorithm. An example of this case is shown below:

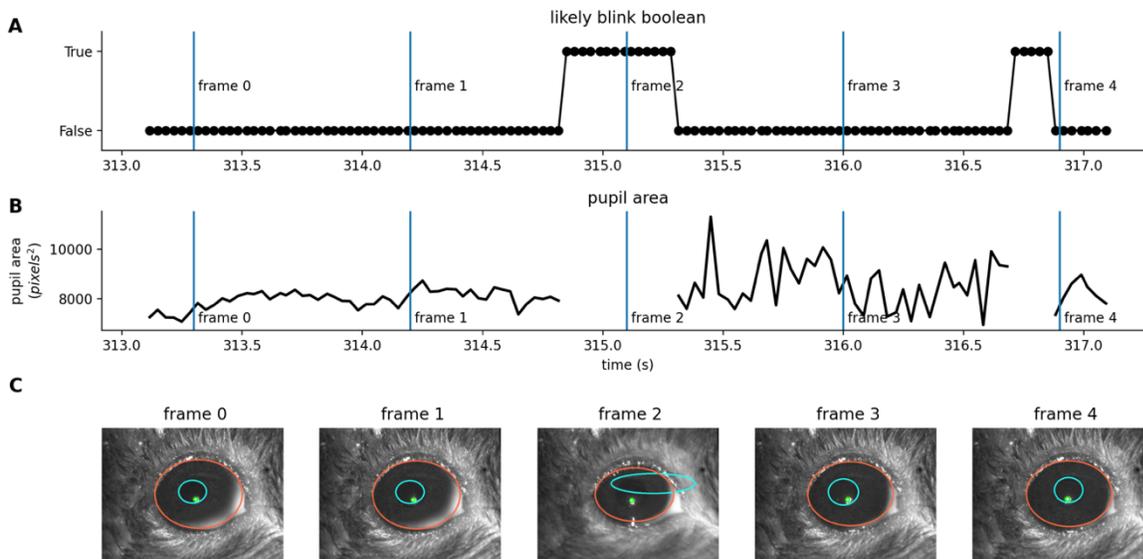


Figure 15. Eye tracking values surrounding an outlier fit. A) The state of the 'likely_blink' Boolean for every frame. B) The calculated area for the pupil fit on every frame. Note that the area is filled with NaN where likely_blink == True. C. Five sample frames with eye and pupil fits overlaid, for the 5 frames annotated with vertical lines in A and B. Note that, in this case, the animal's eye was not actually closed, but the ellipse fit failed to accurately capture the pupil nonetheless.

Given that the raw area calculations are provided, users are free to re-visit the above filtering steps as they see fit.

REFERENCES

- Akturk S, Gu X, Kimmel M, Trebino R. 2006. Extremely simple single-prism ultrashort-pulse compressor. *Opt Express*. doi:10.1364/oe.14.010101
- Chen T-W, Wardill TJ, Sun Y, Pulver SR, Renninger SL, Baohan A, Schreiter ER, Kerr RA, Orger MB, Jayaraman V, Looger LL, Svoboda K, Kim DS. 2013. Ultrasensitive fluorescent proteins for imaging neuronal activity. *Nature* **499**:295–300. doi:10.1038/nature12354
- de Vries SEJ, Lecoq JA, Buice MA, Groblewski PA, Ocker GK, Oliver M, Feng D, Cain N, Ledochowitsch P, Millman D, Roll K, Garrett M, Keenan T, Kuan L, Mihalas S, Olsen S, Thompson C, Wakeman W, Waters J, Williams D, Barber C, Berbesque N, Blanchard B, Bowles N, Caldejon SD, Casal L, Cho A, Cross S, Dang C, Dolbeare T, Edwards M, Galbraith J, Gaudreault N, Gilbert TL, Griffin F, Hargrave P, Howard R, Huang L, Jewell S, Keller N, Knoblich U, Larkin JD, Larsen R, Lau C, Lee E, Lee F, Leon A, Li L, Long F, Luviano J, Mace K, Nguyen T, Perkins J, Robertson M, Seid S, Shea-Brown E, Shi J, Sjoquist N, Slaughterbeck C, Sullivan D, Valenza R, White C, Williford A, Witten DM, Zhuang J, Zeng H, Farrell C, Ng L, Bernard A, Phillips JW, Reid RC, Koch C. 2020. A large-scale standardized physiological survey reveals functional organization of the mouse visual cortex. *Nat Neurosci*. doi:10.1038/s41593-019-0550-9
- Garrett ME, Nauhaus I, Marshel JH, Callaway EM, Garrett ME, Marshel JH, Nauhaus I, Garrett ME. 2014. Topography and areal organization of mouse visual cortex. *J Neurosci*. doi:10.1523/JNEUROSCI.1124-14.2014
- Giovannucci A, Friedrich J, Gunn P, Kalfon J, Brown BL, Koay SA, Taxidis J, Najafi F, Gauthier JL, Zhou P, Khakh BS, Tank DW, Chklovskii DB, Pnevmatikakis EA. 2019. CalmAn an open source tool for scalable calcium imaging data analysis. *Elife*. doi:10.7554/eLife.38173
- Groblewski PA, Ollerenshaw DR, Kiggins JT, Garrett ME, Mochizuki C, Casal L, Cross S, Mace K, Swapp J, Manavi S, Williams D, Mihalas S, Olsen SR. 2020. Characterization of Learning, Motivation, and Visual Perception in Five Transgenic Mouse Lines Expressing GCaMP in Distinct Cell Populations. *Front Behav Neurosci*. doi:10.3389/fnbeh.2020.00104
- Groblewski PA, Sullivan D, Lecoq J, de Vries SEJ, Caldejon S, L'Heureux Q, Keenan T, Roll K, Slaughterbeck C, Williford A, Farrell C. 2020. A standardized head-fixation system for performing large-scale, in vivo physiological recordings in mice. *J Neurosci Methods*. doi:10.1016/j.jneumeth.2020.108922
- Hyvärinen A. 1999. Fast and robust fixed-point algorithms for independent component analysis. *IEEE Trans Neural Networks*. doi:10.1109/72.761722
- Kalatsky VA, Stryker MP. 2003. New paradigm for optical imaging: Temporally encoded maps of intrinsic signal. *Neuron*. doi:10.1016/S0896-6273(03)00286-1
- Lecoq J, Oliver M, Siegle JH, Orlova N, Koch C. 2020. Removing independent noise in systems neuroscience data using DeepInterpolation. *bioRxiv*.
- Macmillan NA, Creelman CD. 2004. Detection Theory: A User's Guide: 2nd edition, Detection Theory: A User's Guide: 2nd edition. doi:10.4324/9781410611147
- Madisen L, Garner AR, Shimaoka D, Chuong AS, Klapoetke NC, Li L, van der Bourg A, Niino Y, Egolf L, Monetti C, Gu H, Mills M, Cheng A, Tasic B, Nguyen TN, Sunkin SM, Benucci A, Nagy A, Miyawaki A, Helmchen F, Empson RM, Knöpfel T, Boyden ES, Reid RC, Carandini M, Zeng H. 2015. Transgenic mice for intersectional targeting of neural sensors and effectors with high specificity and performance. *Neuron*. doi:10.1016/j.neuron.2015.02.022

Marshall JH, Garrett ME, Nauhaus I, Callaway EM. 2011. Functional specialization of seven mouse visual cortical areas. *Neuron*. doi:10.1016/j.neuron.2011.12.004

Oommen BS, Stahl JS. 2008. Eye orientation during static tilts and its relationship to spontaneous head pitch in the laboratory mouse. *Brain Res*. doi:10.1016/j.brainres.2007.11.053

Orlova N, Tsyboulski D, Najafi F, Seid S, Kivikas S, Kato I, Griffin F, Leon A, L'heureux Q, North K, Swapp J, Nayan C, Hancock N, Ahmed R, Gelfand E, Cho A, Mace K, Howard R, Casal L, Lambert S, Lee EK, Caldejon S, Waughman X, Williford A, Garrett M, Ollerenshaw D, Olsen SR, Groblewski P, Lecoq # J, Saggau # P. 2020. Multiplane Mesoscope reveals distinct cortical interactions following expectation violations. *bioRxiv*.

Sofroniew NJ, Flickinger D, King J, Svoboda K. 2016. A large field of view two-photon mesoscope with subcellular resolution for in vivo imaging. *Elife*. doi:10.7554/eLife.14472

Steinmetz NA, Buetfering C, Lecoq J, Lee CR, Peters AJ, Jacobs EAK, Coen P, Ollerenshaw DR, Valley MT, de Vries SEJ, Garrett M, Zhuang J, Groblewski PA, Manavi S, Miles J, White C, Lee E, Griffin F, Larkin JD, Roll K, Cross S, Nguyen T V., Larsen R, Pendergraft J, Daigle T, Tasic B, Thompson CL, Waters J, Olsen S, Margolis DJ, Zeng H, Hausser M, Carandini M, Harris KD. 2017. Aberrant Cortical Activity in Multiple GCaMP6-Expressing Transgenic Mouse Lines. *Eneuro* 4:ENEURO.0207-17.2017. doi:10.1523/ENEURO.0207-17.2017