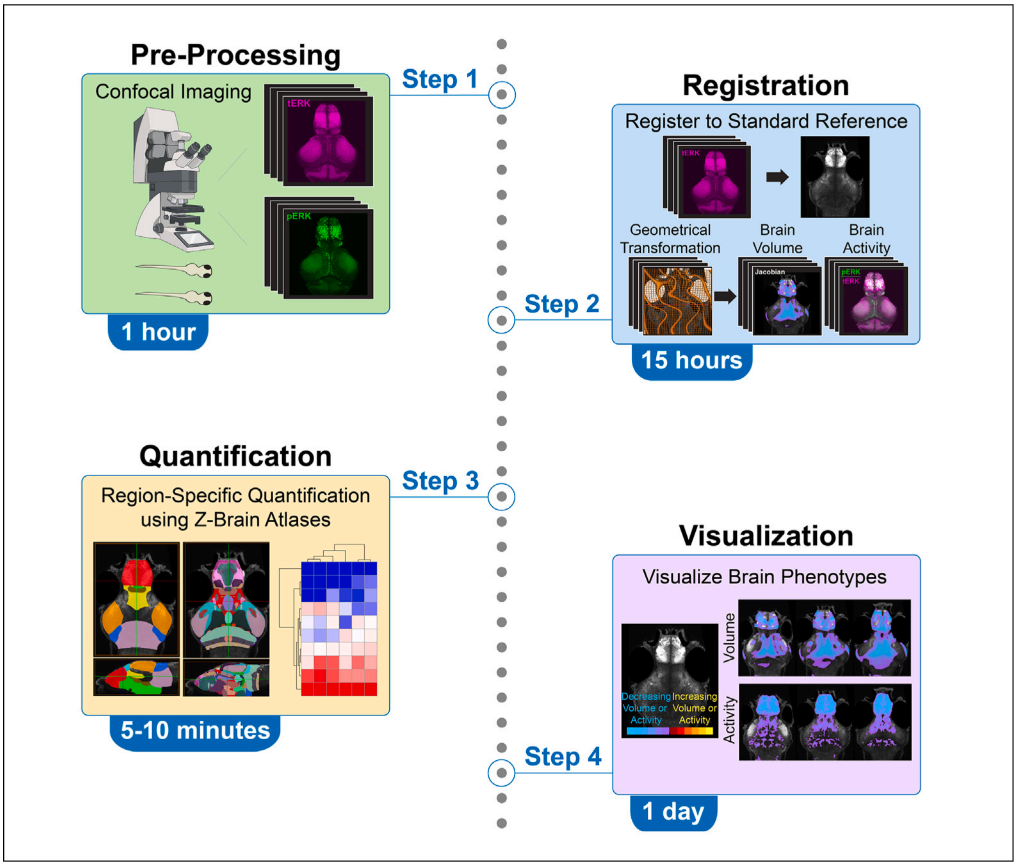
STAR Protocols

CellPress
OPEN ACCESS

Protocol

Brain Registration and Evaluation for Zebrafish (BREEZE)-mapping: A pipeline for whole-brain structural and activity analyses


Pre-Processing
Confocal Imaging
tERK
pERK
1 hour
Step 1
Registration
Register to Standard Reference
Geometrical Transformation
Brain Volume
Brain Activity
Jacobian
pERK
tERK
Step 2
15 hours
Quantification
Region-Specific Quantification using Z-Brain Atlases
Step 3
5-10 minutes
Visualization
Visualize Brain Phenotypes
Volume
Activity
Decreasing Volume or Activity
Increasing Volume or Activity
Step 4
1 day

Here, we present Brain Registration and Evaluation for Zebrafish (BREEZE)-mapping, a user-friendly pipeline for the registration and analysis of whole-brain images in larval zebrafish. We describe steps for pre-processing, registration, quantification, and visualization of whole-brain phenotypes in zebrafish mutants of genes associated with neurodevelopmental and neuropsychiatric disorders. By utilizing BioImage Suite Web, an open-source software package originally developed for processing human brain imaging data, we provide a highly accessible whole-brain mapping protocol developed for users with general computational proficiency.

Publisher's note: Undertaking any experimental protocol requires adherence to local institutional guidelines for laboratory safety and ethics.

David S. Jin, Uma Neelakantan, Cheryl M. Lacadie, ..., Zuoheng Wang, Xenophon Papademetris, Ellen J. Hoffman

ellen.hoffman@yale.edu

Highlights

High-throughput pipeline for the analysis of whole-brain images in larval zebrafish

Tools for identifying quantifiable brain phenotypes in multiple zebrafish mutant lines

Quantification of volume and activity phenotypes using zebrafish brain atlases

User-friendly interface for the 3D visualization of brain phenotypes

Jin, Neelakantan, et al., STAR Protocols 4, 102647
December 15, 2023 © 2023 The Author(s).
https://doi.org/10.1016/j.xpro.2023.102647

STAR Protocols

CellPress
OPEN ACCESS

Protocol

Brain Registration and Evaluation for Zebrafish (BREEZE)-mapping: A pipeline for whole-brain structural and activity analyses

David S. Jin,[1,2,3,4,9,10] Uma Neelakantan,[1,9,10] Cheryl M. Lacadie,[5] Tianying Chen,[1] Brendan Rooney,[1] Yunqing Liu,[6] Weimiao Wu,[6] Zuoheng Wang,[6] Xenophon Papademetris,[5,7,8] and Ellen J. Hoffman[1,2,11,*]

[1]Child Study Center, Yale School of Medicine, New Haven, CT 06510, USA

[2]Department of Neuroscience, Yale University, New Haven, CT 06510, USA

[3]Interdepartmental Neuroscience Program, Yale School of Medicine, New Haven, CT 06510, USA

[4]Department of Neurology, Yale School of Medicine, New Haven, CT 06510, USA

[5]Department of Radiology and Biomedical Imaging, Yale School of Medicine, New Haven, CT 06510, USA

[6]Department of Biostatistics, Yale School of Public Health, New Haven, CT 06510, USA

[7]Section of Biomedical Informatics and Data Science, Yale School of Medicine, New Haven, CT 06510, USA

[8]Department of Biomedical Engineering, Yale University, New Haven, CT 06510, USA

[9]These authors contributed equally

[10]Technical contact

[11]Lead contact

*Correspondence: ellen.hoffman@yale.edu

https://doi.org/10.1016/j.xpro.2023.102647

SUMMARY

Here, we present Brain Registration and Evaluation for Zebrafish (BREEZE)-mapping, a user-friendly pipeline for the registration and analysis of whole-brain images in larval zebrafish. We describe steps for pre-processing, registration, quantification, and visualization of whole-brain phenotypes in zebrafish mutants of genes associated with neurodevelopmental and neuropsychiatric disorders. By utilizing BioImage Suite Web, an open-source software package originally developed for processing human brain imaging data, we provide a highly accessible whole-brain mapping protocol developed for users with general computational proficiency.

For complete details on the use and execution of this protocol, please refer to Weinschutz Mendes et al. (2023).[1]

BEFORE YOU BEGIN

The zebrafish has emerged as a powerful system for investigating basic mechanisms of vertebrate brain development with translational relevance to neurodevelopmental and neuropsychiatric disorders.[2–6] Given the ability to rapidly generate zebrafish mutants in genes of interest using CRISPR/Cas9,[7–10] transparent larvae that allow for visualization of whole-brain development, and a high degree of conservation of neural cell types and circuits controlling basic behaviors in zebrafish and mammals,[11–15] zebrafish are increasingly being utilized as a high-throughput system for analyzing the function of risk genes in the developing brain.

There are a growing number of tools and techniques that have recently been developed to facilitate the analysis of whole-brain phenotypes in zebrafish, including zebrafish brain atlases, such as the Z-Brain atlas,[16] which annotates 294 neuroanatomical regions and multiple neural cell types. In addition, whole-brain activity mapping, or mitogen activated protein kinase (MAP)-mapping, allows for the rapid visualization of regional differences in neural activity in the brains of freely swimming larvae without the need for *in vivo* imaging. Specifically, this technique involves immunostaining whole brains for phospho-extracellular signal-related kinase (pERK), which labels active neurons, and total ERK (tERK), which labels all neural cells. The ratio of pERK to tERK represents a readout of neural activity prior to fixation.[16] This technique has been used by our group and others to identify brain structural and activity phenotypes in mutants of genes associated with autism spectrum disorders (ASDs)[1] and schizophrenia.[17] However, a challenge that limits the widespread use of whole-brain mapping analyses in zebrafish is the considerable computational expertise required for image registration and phenotypic evaluation. Therefore, we developed a highly accessible method for zebrafish whole-brain mapping that requires only general computational proficiency and provides a user-friendly interface for visualizing brain size and activity phenotypes in zebrafish risk gene mutants.

Here, we describe Brain Registration and Evaluation for Zebrafish (BREEZE)-mapping, a pipeline for the analysis and visualization of zebrafish whole-brain imaging datasets. Our group developed this approach to investigate brain size and activity phenotypes in zebrafish mutants of ASD-associated genes, leading to the identification of novel phenotypes across mutants.[1] This pipeline involves the application of tools originally developed for processing human functional magnetic resonance imaging (fMRI) studies to the larval zebrafish brain. Specifically, our method utilizes the open-source BioImage Suite Web[18] software package to perform image registration, region of interest-based quantification, and the 3D visualization of phenotypes. In addition, we use recently developed tools, including a zebrafish standard reference brain (Z-Brain) and the Z-Brain atlas,[16] to quantify regional differences in brain volume and activity in zebrafish larvae at 6 days post fertilization (dpf). While we focus on the analysis of brain volume and activity datasets, our pipeline can also be applied to analyze other cell-specific markers. To ensure accessibility for users with basic computational experience, we provide a step-by-step description of the BREEZE-mapping protocol along with multiple bash scripts (*.sh*) that facilitate implementation. Applications of the pipeline include examining brain phenotypes in zebrafish mutants of neurodevelopmental and neuropsychiatric disorder-associated genes or the effects of pharmacological treatments and environmental exposures on vertebrate brain development, though our pipeline is optimized for studies of genetic mutants. Sample images from zebrafish mutants of ASD-associated genes described in Weinschutz Mendes et al. (2023) are provided.[1] Taken together, BREEZE-mapping represents a streamlined approach for the analysis of larval zebrafish whole-brain imaging datasets that is readily accessible to the broader zebrafish neuroscience community.

Institutional permissions

All procedures involving zebrafish were conducted in accordance with Institutional Animal Care and Use Committee (IACUC; Protocol #2021-20054) regulatory standards at Yale University. Users will need to acquire permissions from their institutions prior to performing these experiments.

Image acquisition

Timing: ∼4–5 days for immunostaining and imaging

Note: The timing described in this protocol is based on a typical experimental dataset, which includes three genotypes (wild-type, homozygous, heterozygous) and ∼10 fish per genotype.

1. Perform immunostaining of larval zebrafish brains at 6 days post fertilization (dpf) as described in previous studies.[1,16,17]
   a. Label brains with an antibody to tERK as a uniform label for best registration results, because the standard reference brain[16] is stained with tERK.

b. Brains may be co-stained with an antibody to pERK for activity mapping or another cell type-specific marker of interest.

⚠ CRITICAL: We strongly recommend using background-matched (sibling- or cousin-matched) homozygous, heterozygous, and wild-type fish to control for genetic background variation, which may affect brain size and baseline activity.

Note: We recommend dissecting larval brains prior to immunostaining to improve antibody penetration and visualization of ventral regions in whole-brain images. The atlas files provided are optimized for use with dissected brains.

Note: When quantifying a cell-specific marker other than pERK, it is important to use an antibody that clearly stains individual cells (i.e., a nuclear stain is preferred) to improve the accuracy of registration.

2. Acquire whole-mount confocal images of immunostained larval zebrafish brains.
   a. Mount larval brains in 1%–2% low-melt agarose and image dorsally at 25X magnification.
   b. Acquire z-stacks in the dorsal to ventral direction for registration, beginning prior to the appearance of the most dorsal brain regions (e.g., optic tectum) and ending after the most ventral regions (e.g., hypothalamus). Z-stacks should span the entire zebrafish brain and range from ∼200−300 μm in depth. The Z-step size is 1.5 μm.
   c. Immunostained brains were imaged at a resolution of 512 × 512 on Leica SP5 or SP8 confocal microscopes in our lab, though no specific confocal microscope is required.
   d. Use the following naming convention for all confocal imaging files: *gene_exp#_genotype_fish#*. An example of a microscope image file name is: *scn1lab_exp1_het_3.lif*.
      i. The pre-processing scripts are based on Leica files (e.g., *.lif*), but can be adapted to other microscope files (e.g., Zeiss, *.czi*).
      ii. The naming convention for genotypes is homozygous (hom), heterozygous (het), or wild-type (wt), which is used throughout the BREEZE-mapping pipeline.

⚠ CRITICAL: The scripts used in the major steps below will only recognize files named using this convention.

Download custom BREEZE-mapping code

⏱ Timing: 4 h

3. Access and download custom BREEZE-mapping code. (https://github.com/ehoffmanlab/Jin_Neelakantan_BREEZE_Mapping; https://doi.org/10.5281/zenodo.7996403).
   a. Unzip the file containing the following three sub-folders:
      i. **pre_processing**: This folder contains scripts that will be run locally to prepare the images for analysis and includes the following four optional codes: ***max_nrrd_2.ijm***; ***max_nrrd_3.ijm***; ***summary_2.js***; ***summary_3.js***.
      ii. **bin**: This folder contains scripts and files that will run on a server or High Performance Computing Cluster (HPCC).
      iii. **test**: This folder contains sample datasets for testing the pipeline.
   b. The **bin** folder contains eight primary scripts that will perform the following functions:
      i. Image registration and processing (***reg.sh***).
      ii. Activity/cell marker intensity quantification (***roi.sh***).
      iii. Volume quantification (***vol.sh***).
      iv. Z-score calculation (***zscore.sh***).
      v. Statistical analysis of activity or cell-specific marker intensity visualization (***afni_roi.sh***).
      vi. Statistical analysis of volume visualization (***afni_vol.sh***).

   - vii. Cluster filtering significant voxel clusters (***clusterfilter.sh***).
   - viii. Optional script for generating custom atlases (***zbrain_universal_script.m***).

- c. The **bin** folder also includes the following reference files, which are required by the above scripts:
   - i. Parameter file (*nonlinearRegistration.param*).
   - ii. Standard zebrafish reference brain files: original (*standard_ref.nii.gz*); resampled (*standard_ref_rsp.nii.gz*).
   - iii. Zebrafish brain atlases (original/resampled): 4-region (*Zbrain_atlas_4region.nii.gz*, *Zbrain_atlas_4region_rsp.nii.gz*); 8-region (*Zbrain_atlas_8region.nii.gz*, *Zbrain_atlas_8region_rsp.nii.gz*); 149-region (*Zbrain_atlas_149region.nii.gz*).
   - iv. Zebrafish brain atlas header files (original/resampled): 4-region (*roi_4region_header.csv*, *vol_4region_header.csv*); 8-region (*roi_8region_header.csv*, *vol_8region_header.csv*); 149-region (*roi_149region_header.csv*).
   - v. Jacobian mask files (*dis_mask_atlas.nii.gz*).
   - vi. AFNI mask files (*Zbrain_binary_mask_rsp_roi.nii.gz*, *Zbrain_binary_mask_rsp.nii.gz*).
- d. The **test** folder includes the following sample datasets:
   - i. Confocal z-stack images of the brains of *scn1lab*$^{+/+}$, *scn1lab*$^{\Delta 44/+}$, and *scn1lab*$^{\Delta 44/\Delta 44}$ fish, which have a robust brain activity phenotype, from one experiment of dissected brains stained for pERK and tERK at 6 dpf.
   - ii. Confocal z-stack images of the brains of *dyrk1aa*$^{+/+}$*dyrk1ab*$^{+/+}$, *dyrk1aa*$^{\Delta 77/+}$*dyrk1ab*$^{\Delta 8/+}$, and *dyrk1aa*$^{\Delta 77/\Delta 77}$*dyrk1ab*$^{\Delta 8/\Delta 8}$ fish, which have a robust brain size phenotype, from one experiment of dissected brains stained for pERK and tERK at 6 dpf.

Note: After downloading the folders containing the custom scripts, move the bin folder to '/usr/local/' or a preferred location. If the location is not '/usr/local/,' the path of the bin directory must be edited.

Note: All scripts are written based on the assumption that there are three genotypes (wt, het, hom) and compare phenotypes to "wild-type" (wt) as the control group. For analyses involving different comparison groups, editing the nomenclature in the scripts is required.

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Antibodies		
Anti-phosphorylated-Erk (1:500)	Cell Signaling	Cat# 4370; RRID: AB_2315112
Anti-total-Erk (1:500)	Cell Signaling	Cat# 4696; RRID: AB_390780
Chemicals, peptides, and recombinant proteins		
BSA	Jackson ImmunoResearch	Cat# A9647-100G
NGS	Jackson ImmunoResearch	Cat# 005-000-121
Paraformaldehyde	Millipore Sigma	Cat# P6148-500G
Trypsin EDTA, 0.5 M solution, pH 8.0	Fisher Scientific	Cat# 15-400-054
Low-melt agarose	American Bioanalytical	Cat# AB00981-000050
Experimental models: Organisms/strains		
Zebrafish (*Danio rerio*): 10 mutant lines Zebrafish larvae at 6 days post fertilization were used in these experiments. Zebrafish are sexually undifferentiated at this stage.	Weinschutz Mendes et al.[1]; Kroll et al.[8]; Hoffman et al.[19]	*chd8*$^{\Delta 5}$, *cul3a*$^{\Delta 7}$*cul3b*$^{\Delta 20}$; *dyrk1aa*$^{\Delta 77}$*dyrk1ab*$^{\Delta 8}$; *grin2ba*$^{\Delta 25}$*grin2bb*$^{\Delta 64}$; *katnal2*$^{\Delta 4}$; *kdm5ba*$^{\Delta 17}$ *kdm5bb*$^{\Delta 14}$; *pogza*$^{\Delta 23}$*pogzb*$^{\Delta 20}$; *tbr1a*$^{\Delta 64}$*tbr1b*$^{\Delta 10}$ mutants are described in Weinschutz Mendes et al.[1]; *scn1lab*$^{\Delta 44}$ mutants are described in Kroll et al.[8] and Weinschutz Mendes et al.[1] *cntnap2a*$^{\Delta 121}$*cntnap2b*^{31ins} mutants are described in Hoffman et al.[19]
Software and algorithms		
MATLAB	MathWorks	https://www.mathworks.com/products/trials.html/

(*Continued on next page*)

Continued

REAGENT or RESOURCE	SOURCE	IDENTIFIER
BioImage Suite (web app)	BioImage Suite Web	bioimagesuiteweb.org
BioImage Suite (software package)	BioImage Suite Web	https://github.com/bioimagesuiteweb/bisweb
biswebnode (software package)	BioImage Suite Web	https://www.npmjs.com/package/biswebnode
Brain mapping code	This paper	https://github.com/ehoffmanlab/Jin_Neelakantan_BREEZE_Mapping; https://doi.org/10.5281/zenodo.7996403
GraphPad Prism	GraphPad	https://www.graphpad.com
ImageJ	NIH	https://imagej.nih.gov/ij/download.html
Adobe Illustrator	Adobe	https://www.adobe.com/products/illustrator.html
FSL	N/A	https://fsl.fmrib.ox.ac.uk/fsl/fslwiki/FslInstallation/MacOsX
Analysis of Functional NeuroImages (AFNI)	NIH	https://afni.nimh.nih.gov/pub/dist/doc/htmldoc/background_install/main_toc.html
NodeJS	N/A	https://nodejs.org/en/
R and RStudio	CRAN	https://www.Rstudio.com
Other		
SP5 and SP8 confocal microscopes	Leica Microsystems	https://www.leica-microsystems.com/

MATERIALS AND EQUIPMENT

- Zebrafish whole-brain (z-stack) images labeled with tERK and another cell-specific marker (e.g., pERK) at 6 dpf obtained by confocal microscopy.
- Linux or MacOS operating system. No specific computer configuration is required. Validated systems used by our lab include a 3.3 GHz Quad-Core Intel i5 iMac with macOS version 12.6.8.
- High-performance computing cluster (HPCC) or data-processing server. Timing estimates in this protocol are based on analysis using the HPCC at Yale University, which is comprised of 300 nodes with 12,000 10–16GB CPU cores. The HPCC runs Linux as its operating system and uses Slurm as its scheduling mechanism.
- The following software packages are required:
  - MATLAB version 2020b or above*
  - BioImage Suite WebApp (and companion command line tools biswebnode)[18],*
  - FSL*
  - NodeJS*
  - Analysis of Functional Neuroimaging (AFNI) (NIH)[20],*
  - ImageJ (NIH)[21]

> ***Note:*** The software packages labeled with an asterisk (*) above are required on an HPCC to run the BREEZE-mapping pipeline. Most of these packages are available on institutional HPCCs. Please check with your systems administrator to confirm that these packages are installed on your institutional HPCC. Specific BioImage Suite command line tools are installed on the HPCC through the node.js package manager.

STEP-BY-STEP METHOD DETAILS

Here, we provide a step-by-step approach to BREEZE-mapping. We describe in detail the following four major steps (Figure 1, Table S1): *(1)* ***Pre-processing:*** This step converts the image files to the format required for registration; *(2)* ***Registration:*** This step involves the non-linear registration[22] of larval zebrafish whole-brain z-stack images labeled with a uniform marker (e.g., tERK) to a standard zebrafish reference brain[16] using transformation-based deformation,[23,24] followed by re-slicing and registration of other fluorescent channels (e.g., pERK or other cell-specific markers) using the same geometric transformation file; *(3)* ***Quantification:*** In this step, brain volume, activity, or cell-specific markers are quantified within regions of interest designated by zebrafish brain atlases;[16] *(4)* ***Visualization:*** In this step, 3D maps are used to visualize differences in brain volume, activity,

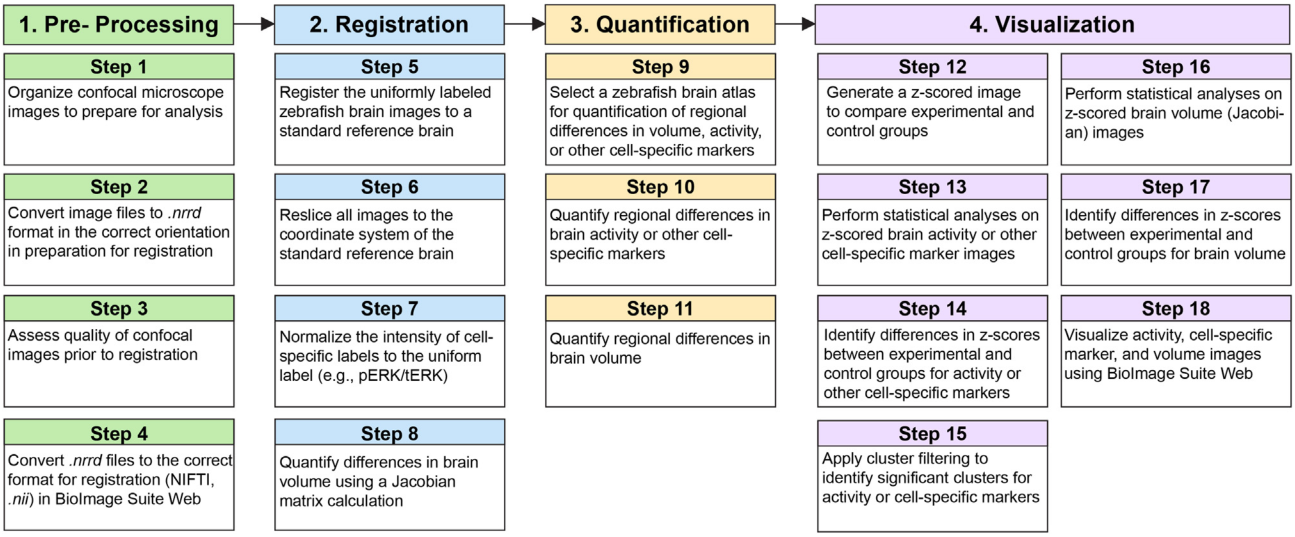

1. Pre- Processing
2. Registration
3. Quantification
4. Visualization
Step 1
Organize confocal microscope images to prepare for analysis
Step 2
Convert image files to .nrrd format in the correct orientation in preparation for registration
Step 3
Assess quality of confocal images prior to registration
Step 4
Convert .nrrd files to the correct format for registration (NIFTI, .nii) in BioImage Suite Web
Step 5
Register the uniformly labeled zebrafish brain images to a standard reference brain
Step 6
Reslice all images to the coordinate system of the standard reference brain
Step 7
Normalize the intensity of cell-specific labels to the uniform label (e.g., pERK/tERK)
Step 8
Quantify differences in brain volume using a Jacobian matrix calculation
Step 9
Select a zebrafish brain atlas for quantification of regional differences in volume, activity, or other cell-specific markers
Step 10
Quantify regional differences in brain activity or other cell-specific markers
Step 11
Quantify regional differences in brain volume
Step 12
Generate a z-scored image to compare experimental and control groups
Step 13
Perform statistical analyses on z-scored brain activity or other cell-specific marker images
Step 14
Identify differences in z-scores between experimental and control groups for activity or other cell-specific markers
Step 15
Apply cluster filtering to identify significant clusters for activity or cell-specific markers
Step 16
Perform statistical analyses on z-scored brain volume (Jacobian) images
Step 17
Identify differences in z-scores between experimental and control groups for brain volume
Step 18
Visualize activity, cell-specific marker, and volume images using BioImage Suite Web

Figure 1. Flowchart of major steps in the BREEZE-mapping pipeline

The pipeline includes the following major steps: 1) Pre-processing; 2) Registration; 3) Quantification; 4) Visualization. In Pre-processing (Steps 1-4), images are organized, oriented, and quality-assessed in preparation for registration. Images then undergo Registration (Steps 5-8), where they are co-registered to a standard zebrafish reference brain image, and normalized intensity of cell-specific markers to the uniform marker and localized brain volume (Jacobian values) are calculated. In Quantification (Steps 9-11), standard zebrafish brain atlases are applied to quantify regional differences in intensity and volume. In Visualization (Steps 12-18), image values are normalized and family-wise error (FWE) correction is applied to visualize statistically significant differences between experimental and control groups. A list of input and output files and scripts utilized at each step are shown in Table S1.

or cell-specific markers between experimental and control groups by applying statistical testing for multiple comparisons. Overall, this protocol provides a user-friendly pipeline for the analysis of whole-brain volume and activity phenotypes in larval zebrafish.

Pre-processing

Timing: 30 min - 1 h

The objective of this major step is to ensure that whole-brain confocal images are in the correct orientation and data format for registration. Specifically, z-stack image files (e.g., *.lif, .czi*) are first converted to *.tif* and *.nrrd* format and then to *.nii* (''*optfixed*'') format.

1. Organize confocal microscope files in preparation for pre-processing. In a folder, create the following three subfolders, which will be used in the major steps below:
   a. **tif** containing all *.tif* files
   b. **nrrd** containing all *.nrrd* files
   c. **optfixed** containing *.nii* (''*optfixed*'') files

⚠ CRITICAL: The optfixed folder name is required because the scripts use ''optfixed'' as a keyword for image location.

2. For quality assessment, visualize maximum projections of all z-stack images prior to registration. Using the ImageJ scripts, convert whole-brain confocal z-stack images (*.lif* files) to max projection files (*.tif*) and converted images (*.nrrd*) in the correct orientation for registration.
   a. Run *max_nrrd_2.ijm* or *max_nrrd_3.ijm* from the **pre_processing** folder. These scripts are written for 2- or 3-channel images, respectively.
   b. To run these scripts in ImageJ:
      i. Open the script (File → Open → *max_nrrd_2.ijm or max_nrrd_3.ijm* → Run).

c. Input folder locations of images as prompted in the following sequence: **lif** (location of *.lif* files) → **tif** (folder for max projection *.tif* files to be created) → **nrrd** (folder for *.nrrd* files to be created).
d. The script will generate a max projection image and the user will be prompted to rotate the image to the optimal orientation for image visualization, which is anterior (forebrain) facing up.
   i. The script provides a drop-down menu with options to rotate the image: 0 = no rotation; 1 = 90 degrees clockwise; 2 = 180 degrees clockwise; 3 = 270 degrees clockwise.
e. A *.nrrd* file will simultaneously be generated 90 degrees clockwise to the orientation of the *.tif*. This file will be in the correct orientation for registration, which is anterior facing right. It is important to verify that the acquisition orientation is correct for your dataset/microscope.

⚠ CRITICAL: The scripts are written to account for C0, C1, and C2 as keywords, where "C1" represents the uniform neuronal label (i.e., tERK), and "C0" and "C2" are different cell-specific markers. Z-stack files must be named according to this convention (e.g., *gene_exp#_genotype_fish#_C0.nrrd*, *gene_exp#_genotype_fish#_C1.nrrd*).

3. Visualize all confocal images as max projections to assess image quality. Scripts for use in Adobe Illustrator are provided to facilitate this process.
   a. Run ***summary_2.js***, ***summary_3.js*** from the **pre-processing** folder. These scripts are written for 2- or 3-channel images, respectively.
   b. To run these codes in Adobe Illustrator:
      i. Create a new template (any size template may be used; Color Mode: RGB).
      ii. Open the script (File → Scripts → Other Scripts).
      iii. The script will prompt the user to open the **tif** folder containing the max projection files.
      iv. The script will automatically display the max projection images organized by genotype on the template.

⚠ CRITICAL: It is important to check image quality and exclude any samples with obvious inconsistencies in dissecting, staining, or mounting. Excessively rotated (i.e., roll, pitch, and yaw relative to the standard reference brain), damaged, or inconsistently stained samples may cause unnatural warping of images, resulting in inaccurate registration. In such cases, it may be necessary to exclude a sample from analysis.

4. Convert whole-brain z-stack images from *.nrrd* to *.nii* ("optfixed") format using BioImage Suite Web.
   a. Open the BioImage Suite Web App: https://bioimagesuiteweb.github.io/webapp/viewer.html#
   b. In Bioimage Suite, navigate to "Help" and select "Enable Internal Feature." Once enabled in a given browser, this step is no longer required for subsequent analyses.
   c. Navigate to "Image Processing" and select "Fix Zebrafish Images."
   d. Using the side bar, set the following parameters: Select *RAS* and *resample*; unselect *biascorrect, normalize*, and *mask*; set Smooth sigma = 1 and Resample Factor = 1.
   e. Load each *nrrd* file individually in Bioimage Suite Web, select "Execute," and save each file. Images will be saved as **optfixed** images (*_optfixed.nii.gz*).
   f. In preparation for registration, upload the **optfixed** folder containing all converted images (*_optfixed.nii.gz*) to a dedicated **experiment** folder on an HPCC.

Note: It is not recommended to automate this step because it serves as a checkpoint for image orientation and allows users to visualize the converted images prior to registration.

Note: This is the transition point from processing files locally to running analyses on an HPCC.

Registration

⏲ Timing: ~15 h

The objective of this major step is to standardize the geometry of imaged zebrafish brains to a standard reference brain image. This step generates a geometric transformation file for each image labeled with a uniform marker (e.g., tERK), which is used to register other channels from the same sample to the standard reference and to calculate volume differences. For voxel-based quantification of a cell-specific marker, the intensity of the cell-specific marker is divided by the uniform/pan-neuronal marker (e.g., pERK/tERK) for normalization.

⚠ CRITICAL: For the following steps, please ensure that the scripts and software packages listed in "materials and equipment" above with an asterisk (*) are installed on the HPCC.

5. In a bash terminal, execute the script, ***reg.sh***. This will register the uniformly labeled zebrafish brain images (*_optfixed.nii.gz*) to a standard zebrafish reference brain.[16]
   a. To execute ***reg.sh***, change directories to the **experiment** folder containing the **optfixed** folder. In the command line, enter ***bash reg.sh***.
      i. Successful execution of ***reg.sh*** will generate a list of submitted registration jobs.
   b. Input: specify the following parameters within ***reg.sh***:

```
BINDIR="/usr/local/bin"

STANDARD REFERENCE="standard ref.nii.gz"

STANDARD_RSP_REFERENCE= "standard_ref_rsp.nii.gz"

NON_PARAM_FILE="nonlinearRegistration.param"

DIS_ATLAS_MASK_FILE="dis_atlas_mask.nii.gz"
```

      i. BINDIR: Path of bin directory containing all files and scripts.
      ii. STANDARD_REFERENCE: Standard zebrafish reference brain. This allows the dataset to be registered and resampled to a standard reference coordinate system. *standard_ref.nii.gz,* derived from Randlett et al. (2015),[16] is the default reference image used in the pipeline.
      iii. STANDARD_RSP_REFERENCE: Downsampled version of the standard zebrafish reference brain. This is used to generate the determinant of the Jacobian image (subsequently referred to as the Jacobian). *standard_ref_rsp.nii.gz* is the default resampled image.
      iv. NON_PARAM_FILE: This file (*nonlinearregistration.param*) contains a set of parameters for image registration to be interpreted by biswebnode. The biswebnode command line tools are available via the node package manager (npm) of node.js.
      v. DIS_ATLAS_MASK_FILE: This file (*dis_atlas_mask.nii.gz*) is a binary indication of voxels corresponding to the standard zebrafish reference brain, which is used as a "dissected" mask for the Jacobian file.

⚠ CRITICAL: The location of the bin folder must be specified in *reg.sh*. Please ensure that the bin folder is uploaded to the HPCC.

   c. Output: ***reg.sh*** generates an intermediate *.bisxform* file for each uniformly labeled (i.e., C1, tERK) brain image containing the information necessary to register all channels of the same image to the reference brain.
      i. These files will appear in a new **xform** subfolder in the following format: *_C1__optfixed__nlr.bisxform*.

```
biswebnode nonlinearregistration --paramfile ${NON_PARAM_FILE} -r ${STANDARD_REFERENCE} -t
\${input_cl_file}
```

⏸ **Pause point:** Using an HPCC will allow multiple jobs to run simultaneously. Each job is an image from one fish. After jobs are submitted, we recommend checking their status to ensure that jobs are running appropriately.

6. Reslice images with cell-specific labels to the coordinate system of the standard reference.
   a. The script, ***reg.sh*** applies the geometric transformation file (*.bisxform*) to both the uniformly labeled brain image in the C1 channel (tERK) and the cell-specific marker(s) in the C0 and C2 channels. The script uses the ***resliceimage*** command in BioImage Suite to reslice the image in each channel and align it with the standard reference brain.
   b. Input:

```
input_c0_file="optfixed/${file}_C0_optfixed.nii.gz"

input_cl_file="optfixed/${file}_Cl_optfixed.nii.gz"

input_c2_file="optfixed/${file}_C2_optfixed.nii.gz"
```

      i. input_C0_file (*_C0__optfixed.nii.gz): Pre-processed first cell-specific marker image.
      ii. input_C1_file (*_C1__optfixed.nii.gz): Pre-processed uniform marker image.
      iii. input_C2_file (*_C2__optfixed.nii.gz): Pre-processed second cell-specific marker image.
   c. Output: The registered files for all channels will appear in a new **registered** subfolder in the following format: r_*_C0__optfixed.nii.gz, r_*_C1__optfixed.nii.gz.

```
biswebnode resliceimage -r ${STANDARD_REFERENCE} -i \${input_cl_file} -x \${bisxform_file} -o
\${registered_cl_file}

biswebnode resliceimage -r \${registered_cl_file} -i\${input_c0_file} -x \${bisxform_file} -o
\${registered_c0_file}
```

7. To determine regional differences in the cell-specific label (e.g., pERK for active neurons), normalize the intensity of the cell-specific label to the uniform label (e.g., tERK for total neurons). This step is performed by the script, ***reg.sh.***
   a. ***reg.sh*** divides the three-dimensional whole-brain z-stacks labeled with the cell-specific marker in the C0 or C2 channels with those labeled by the uniform marker in the C1 channel. ***reg.sh*** uses the ***-div*** option in the ***fslmaths*** command to divide the intensity value of each voxel in the cell-specific image with the intensity value of the corresponding voxel in the uniformly labeled image.

```
registered_c0_file="r_${file}_CO_optfixed.nii.gz"

registered cl_file="r_${file}_Cl_optfixed.nii.gz"

registered_c2_file="r_${file}_C2_optfixed.nii.gz"
```

   b. Input:
      i. registered_c0_file (r_*_C0__optfixed.nii.gz): Registered first cell-specific marker image.
      ii. registered_c1_file (r_*_C1__optfixed.nii.gz): Registered uniform marker image.
      iii. registered_c2_file (r_*_C2__optfixed.nii.gz): Registered second cell-specific marker image.
   c. Output: This step generates a new file for each image (r_*C0divC1.nii.gz) representing the voxel-by-voxel division of the two z-stacks. These files will appear in a new **divided** subfolder, which contains subfolders for each cell-specific marker (e.g., **C0divC1**, **C2divC1**).

```
mkdir -p divided/C0divC1

divided_file="r_${file}_C0divC1.nii.gz"

fslmaths \${registered_c0_file} -div \${registered_c1_file}

divided/C0divC1/\${divided_file}

mkdir -p divided/C2divC1

divided_file ="r_${file}_C2divC1.nii.gz"

fslmaths \${registered_c2_file} -div \${registered_c1_file}

divided/C2divC1/\${divided_file}
```

Note: As described above, the scripts are written to account for C0, C1, and C2 as keywords, where "C1" represents the uniform neuronal label (tERK), and "C0" and "C2" are two different cell-specific markers. If only one cellular marker is used, the script will only generate calculations for the image channel that is used (e.g., C0 or C2).

8. To quantify differences in brain volume, perform a Jacobian calculation using the *.bisxform* transformation files (*standard_ref*C1_optfixed_nlr.bisxform*). This file describes the degree of stretch or compression required to map the test brain to the standard reference brain. This step is performed by the script, ***reg.sh.***
   a. By using the ***jacobianimage*** function of the BioImage Suite package, ***reg.sh*** calculates the ratio between the test brain and a 2 × 2 × 2 resampled version of the standard reference brain. This ratio reflects whether the test brain is larger or smaller than the reference in specific regions (i.e., 1 = no volume change, > 1 = individual subject is larger than the template and < 1 = individual subject is smaller than the template).[23]
   b. Input:
      i. STANDARD_RSP_REFERENCE: Downsampled version of the standard zebrafish reference brain. This is used to generate the Jacobian. *standard_ref_rsp.nii.gz* is the default resampled reference brain image.
      ii. DIS_ATLAS_MASK_FILE: This file (*dis_atlas_mask.nii.gz*) is a binary indication of voxels corresponding to the standard zebrafish reference brain, which is used as a "dissected" mask for the Jacobian file.
   c. Output: The Jacobian calculation generates the following files for each uniformly-labeled (i.e., C1, tERK) image: *ja_*.nii.gz* and *mask_ja*.nii.gz*, which will appear in a new **jacobian** subfolder.

```
biswebnode jacobianimage -x \${bisxform_file} -i

${BINDIR}/${STANDARD_RSP_REFERENCE} -o jacobian/\${jacobian_file}

biswebnode maskimage -i jacobian/\$ {jacobian_file} -m

${BINDIR}/${DIS_ATLAS_MASK_FILE} -o jacobian/\${mask_file}
```

Note: The Jacobian image file contains all values irrespective of the location in the fish. To include only values corresponding to the fish brain, the *mask_ja*.nii.gz* file is generated using a binary mask file corresponding only to regions of the Z-Brain atlas within the brain.

⚠ CRITICAL: Because Jacobian calculations are performed on the geometric transformation files, which are a different dimensionality compared to the registered images, these calculations require the use of a 2 × 2 × 2 resampled version of the zebrafish standard reference brain (*standard_ref_rsp.nii.gz*). To confirm that the image and the standard reference have the same dimensions, use the *headerinfo* tool by typing *biswebnode headerinfo* in terminal followed by the name of the image. This will output the x,y,z dimensions of the image as well as the orientation space (right or left, anterior or posterior, superior or inferior).

Note: Checkpoint: Confirm that ***reg.sh*** generated the following subfolders containing the expected files: *(i)* **xform:** files containing the mapping calculations for the non-linear registration of each image (*.bisxform*); *(ii)* **registered:** registered versions of each channel to the standard reference for each image (*r_*_C0__optfixed.nii.gz*, *r_*_C1__optfixed.nii.gz*); *(iii)* **divided:** normalized files (*r_*C0divC1.nii.gz, r_*C2divC1.nii.gz*); *(iv)* **jacobian:** the Jacobian files (*ja_*.nii.gz*, *mask_ja*.nii.gz*); and *(v)* **logs**. The logs subfolder provides a record of the jobs run and any errors encountered.

Quantification

Timing: 5–10 min

The objective of this major step is to quantify regional image intensities representing brain activity (pERK) or other cell-specific markers and brain volume across multiple fish in each experimental group using zebrafish brain atlases to define regions of interest.

9. To quantify regional differences in brain size and/or activity, we provide multiple versions of the zebrafish brain atlas adapted from Randlett et al. (2015)[16] and Thyme et al. (2019)[17] (as described in Weinschutz Mendes et al. (2023)[1]).
   a. **4-region** (*Zbrain_atlas_4region.nii.gz*, *Zbrain_atlas_4region_rsp.nii.gz*): An atlas for quantifying phenotypes across four major subdivisions of the zebrafish brain: *(i)* telencephalon; *(ii)* diencephalon; *(iii)* mesencephalon; *(iv)* rhombencephalon.
   b. **8-region** (*Zbrain_atlas_8region.nii.gz*, *Zbrain_atlas_8region_rsp.nii.gz*): An atlas for quantifying phenotypes across eight major brain regions: *(i)* forebrain (telencephalon); *(ii)* optic tectum; *(iii)* thalamus; *(iv)* hypothalamus; *(v)* cerebellum; *(vi)* hindbrain; *(vii)* habenula; *(viii)* posterior tuberculum.
   c. **149-region** (*Zbrain_atlas_149region.nii.gz*): A 149-region atlas derived from the Z-Brain atlas developed by Randlett et al. (2015),[16] excluding the external features or ganglia. This atlas is optimized for the analysis of dissected brain images.

Optional: To generate custom-designed atlases, we provide the ***zbrain_universal_script.m***, which allows users to integrate ROIs derived from the Z-Brain atlas to create new atlases for subpopulations of interest.

⚠ CRITICAL: Jacobian calculations require the use of 2 × 2 × 2 resampled versions of zebrafish brain atlases (**_rsp.nii.gz*). The dimensions of these atlases can be verified using the *biswebnode headerinfo* command.

10. To quantify regional differences in pERK (brain activity) or other cell-specific markers, execute the bash script, ***roi.sh*** in the **experiment** folder.
    a. ***roi.sh*** generates *.csv* files containing the average intensity values (pERK/tERK) per brain region depending on the atlas selected. In addition to generating *.csv* files for each fish individually, ***roi.sh*** integrates the normalized intensity values for all fish in the experiment into a combined *.csv* file (**_C0divC1_combined.csv*). To run ***roi.sh***, type ***roi.sh*** in the terminal command line followed by the number designating the desired atlas (e.g., ***roi.sh 4***, ***roi.sh 8***, ***roi.sh 149*** for the four-, eight-, and 149-region zebrafish atlases, respectively, as separate commands).

b. Input:

```
BINDIR="/usr/local/bin"
DIV_TYPE="C0divCl C2divC1"
ATLAS_4="Zbrain_atlas_4region.nii.gz"
ATLAS_8="Zbrain_atlas_8region.nii.gz"
ATLAS_149="Zbrain_atlas_149region.nii.gz"
HEADER_4="roi_4region_header.csv"
HEADER_8="roi_8region_header.csv"
HEADER_149="roi_149region_header.csv"
```

i. BINDIR: Path of bin directory containing all files and scripts.
ii. DIV_TYPE (*r_*C0divC1.nii.gz, r_*C2divC1.nii.gz*): Normalized/divided files generated by ***reg.sh***.
iii. ATLAS_4 (*Zbrain_atlas_4region.nii.gz*): 4-region atlas.
iv. ATLAS_8 (*Zbrain_atlas_8region.nii.gz):* 8-region atlas.
v. ATLAS_149 (*Zbrain_atlas_149region.nii.gz*): 149-region atlas.
vi. HEADER_4 (*roi_4region_header.csv*): Header for the 4-region atlas.
vii. HEADER_8 (*roi_8region_header.csv*): Header for the 8-region atlas.
viii. HEADER_149 (roi_149region_header.csv): Header for the 149-region atlas.

⚠ CRITICAL: The location of the bin folder must be specified in *roi.sh*.

c. Output: This script uses the ***computeroi*** function of BioImage Suite, the files in the **divided** subfolders (*r_*C0divC1.nii.gz, r_*C2divC1.nii.gz*), and a zebrafish brain atlas to calculate the average intensity values of the ratio of the cell-specific marker to tERK within the regions of interest (ROIs) designated by the atlas. These files are deposited in a new **roi_*** subfolder, which is named according to which atlas is used and contains subfolders for each cell-specific marker (e.g., **C0divC1**, **C2divC1**).

```
for file in \$(cd divided/\${div_type} 2> /dev/null; ls
r_*\${div_type}.nii.gz 2> /dev/null)
do
zbrain_file="${BINDIR}/${ATLAS}"
roi_csv_file=\$(echo \$file | sed
’s/.nii.gz/_${region}_roi.csv/g’)
biswebnode computeroi -i divided/\${div_type}/\$file -r
\${zbrain_file} -o \${roi folder}/\${roi_csv_file}
done
```

11. To quantify regional differences in brain volume, execute the bash script, ***vol.sh*** in the **experiment** folder.
    a. ***vol.sh*** generates *.csv* files containing the average Jacobian values for each brain region depending on the brain atlas selected. In addition to generating *.csv* files for each fish individually, ***vol.sh*** integrates the Jacobian values for all fish in the experiment into a combined *.csv*

file (*mask_ja*_combined.csv*). To run ***vol.sh***, type ***vol.sh*** in the terminal command line followed by the number designating the desired atlas (e.g., ***vol.sh 4***, ***vol.sh 8*** for the four- and eight-region zebrafish atlases, respectively, as separate commands).

b. Input:

```
BINDIR="/usr/local/bin"

JACOBIANDIR="jacobian"

ATLAS_4="Zbrain_atlas_4region_rsp.nii.gz"

ATLAS_8= "Zbrain_atlas_8region_rsp.nii.gz"

HEADER_4="vol_4region_header.csv"

HEADER_8="vol_8region_header.csv"
```

i. BINDIR: Path of bin directory containing all files and scripts.
ii. JACOBIANDIR: (*mask_ja*.nii.gz*): Masked Jacobian files generated by ***reg.sh***.
iii. ATLAS_4 (*Zbrain_atlas_4region_rsp.nii.gz*): Resampled 4-region atlas.
iv. ATLAS_8 (*Zbrain_atlas_8region_rsp.nii.gz*): Resampled 8-region atlas.
v. HEADER_4 (*vol_4region_header.csv*): Header for the 4-region atlas.
vi. HEADER_8 (*vol_8region_header.csv*): Header for the 8-region atlas.

⚠ CRITICAL: The location of the bin folder must be specified in *vol.sh*.

⚠ CRITICAL: Due to the downsampling required for volume measurements, we do not recommend using the 149-region atlas when analyzing brain volume, because small ROIs may not accurately reflect regional volume differences. Furthermore, downsampling may remove small regions entirely, such that their activity and volume may not be directly comparable.

c. Output: This script uses the ***computeroi*** function of BioImage Suite, the files in the **jacobian** subfolder (*mask_ja*.nii.gz*), and a resampled zebrafish brain atlas to calculate the Jacobian values within the regions of interest (ROIs) designated by the atlas.

```
for file in \$ (cd ${JACOBIANDIR} 2> /dev/null; ls

mask_ja*.nii.gz 2> /dev/null)

do

zbrain_file= "${BINDIR}/${ATLAS}"

vol csv_file=\$ (echo\$file | sed

’s/.nii.gz/_${region}_vol . csv/g ’)

biswebnode computeroi -i $JACOBIANDIR/\$file -r

\${zbrain_file} -o \${vol_folder}/\${vol_csv_file}

done
```

⚠ CRITICAL: To analyze significant differences in regional brain activity (**_C0divC1_combined.csv*) and volume (*mask_ja*_combined.csv*) between wild-type and mutant groups, we utilized linear mixed models, because this statistical method accounts for variability associated with the day of testing (described below in quantification and statistical analysis).

⏸ **Pause point:** Due to the highly automated nature of the pipeline, quality assessment for individual images may be necessary should statistical outliers arise. BioImage Suite Editor (https://bioimagesuiteweb.github.io/webapp/editor.html) may be used to validate individual values generated by ***roi.sh*** and ***vol.sh***. To do this, first, load each image (*r_*C0divC1.nii.gz* or *mask_ja*.nii.gz*) in BioImage Suite (File → Load Image). Next, load the desired zebrafish brain atlas in the viewer as the "Objectmap" (Objectmap → Load Objectmap). Finally, perform a "VOI Analysis" (Objectmap → VOI Analysis) ("VOI" is used interchangeably with "ROI"), which may be exported as a comma-separated value (*.csv*) file using the "Export as CSV" option. We recommend that images generated in future steps, such as those generated by ***zscore.sh*** and ***afni_roi.sh***, be visualized in the same manner to assess for statistical outliers.

Visualization

⏱ Timing: ∼ 1 day

The objective of this major step is to visualize differences in brain activity, cell-specific markers, or volume between experimental and control groups. To identify significant voxels for brain activity, a 3-D t-test and cluster filtering are performed using the AFNI 3dClustSim version (16.3.05, October 2016) program.[20] These steps generate files and datasets used to cluster filter significant voxels. The BioImage Suite WebApp Viewer Tool is used for image visualization.

12. To generate a z-scored image for either cell-specific markers (e.g., pERK) or volume datasets, execute the script ***zscore.sh*** in the **experiment** folder.
    a. To run the z-score quantification script, type ***zscore.sh*** in the terminal command line followed by a list of the experimental groups and control group.
        i. To compare homozygous (hom) and heterozygous (het) mutants to wild-type (wt) samples, type ***zscore.sh wt het hom***, where the group names correspond to the genotypes in the image file names. The script assumes that "wt" is the control group unless otherwise specified and the group names may be written in any order.
        ii. ***zscore.sh*** incorporates a MATLAB script (***zscore.m***) that uses the ***fslmaths*** command in FSL to generate a z-score image file for all samples in the experimental group(s) that represents the z-score of each voxel with respect to the standard deviation and mean of the control group.
        iii. ***zscore.m*** uses the files in the **divided** subfolder (**_C0divC1.nii.gz*) for brain activity or other cell-specific markers and the files in the **jacobian** subfolder (*mask_ja*.nii.gz*) for brain volume.
        iv. ***zscore.m*** also requires mean (*avg_*_wt.nii.gz*) and standard deviation (*stdev_*_wt.nii.gz*) files for the control group and two text files (**fishlist.txt*) containing a list of the image names in the experimental and control groups. These files are generated automatically by ***zscore.sh*** based on the genotypes specified in the command line.

Note: This is the first point at which the pipeline directly quantifies differences between experimental groups.

    b. Input:

```
FSLDIR=’/usr/local/bin’

BINDIR="/usr/local/bin"

DIV_TYPE="C0divC1 C2divC1"

JACOBIAN_REFERENCE _FISH="${BINDIR}/standard_ref_rsp.nii.gz"

DIVIDED_REFERENCE_FISH="${BINDIR}/standard_ref.nii.gz"
```

i. FSLDIR: Path of directory containing FSL.
ii. BINDIR: Path of bin directory containing all files and scripts.
iii. DIV_TYPE: Normalization type/variable ("C0divC1, C2divC1").
iv. JACOBIAN_REFERENCE_FISH: Resampled standard reference brain (*standard_ref_rsp.nii.gz*).
v. DIVIDED_REFERENCE_FISH: Standard reference brain (*standard_ref.nii.gz*).

⚠ CRITICAL: The location of the bin folder must be specified in *zscore.sh.*

c. Output: The z-score, mean, standard deviation, and text files specifying the experimental groups will appear in two subfolders: **divided_zscore** (activity or cell-specific marker) and **jacobian_zscore** (volume).

```
fslDirectory = '${FSLDIR}'

referenceGroupList = textread ('${referenceGroupList}', '%s')

comparisonGroupList = textread('${comparisonGroupList}', '%s')

meanRefimageName = '${meanRefimageName}'

sdimageName = '${sdimageName}'

referenceFishimage = '${referenceFishimage}';
```

⏸ **Pause point:** For quality assessment, we recommend visualizing z-scored images for each genotype using BioImage Suite Editor (https://bioimagesuiteweb.github.io/webapp/editor.html), as described above.

⚠ CRITICAL: To visualize statistically significant voxel intensities for brain activity or other cell-specific markers, multiple-comparisons corrections are performed in BioImage Suite and AFNI, as described in Steps 13–15 below.

13. To perform statistical testing on the z-scored activity or cell-specific marker images, execute the script ***afni_roi.sh*** in the **experiment** folder followed by a list of the experimental and control groups. In the command line, enter ***bash afni_roi.sh wt het hom***.
    a. Input:

```
PWD=$(basename `pwd`)

BINDIR="/usr/local/bin"

ZBRAIN BINARY FILE="Zbrain_binary_mask_rsp_roi.nii.gz"

DIV_TYPE="C0divCl C2divCl"

GENO_TYPE="$@"

GENO_TYPE="${GENO_TYPE//[wW][tT]/}"
```

i. BINDIR: Path of bin directory containing all files and scripts.
ii. ZBRAIN_BINARY_FILE: Binary mask indicating voxels that are within the fish (*Zbrain_binary_mask_rsp_roi.nii.gz*).
iii. DIV_TYPE: **C0divC1**; **C2divC1** folders.
iv. GENO_TYPE: Genotype (e.g., heterozygous, homozygous, wild-type).

⚠ CRITICAL: The location of the bin folder must be specified in *afni_roi.sh.*

b. *afni_roi.sh* uses the z-score files (*r_*C0divC1_zscore.nii.gz*) and first runs the following steps using BioImage Suite:
   i. Smoothing (*biswebnode smoothimage*): Images are initially smoothed to reduce background noise. Individual z-score files are smoothed by a sigma factor of 3.
   ii. Resampling (*biswebnode resampleimage*): Images are downsampled by a factor of 2 × 2 × 2 to account for the computationally intensive nature of the calculation.
   iii. After the z-score files are smoothed and resampled, they will be placed into subfolders within the folder, afni_roi, distinguished by genotype: hom_smoothed_rsp, het_smoothed_rsp, and wt_smoothed_rsp.
   iv. Concatenation (*biswebnode concatenateimage*): Within each subfolder, *afni_roi.sh* concatenates the smoothed and resampled files and outputs a concatenated file (*_*concatenation.nii.gz*) for each genotype in the respective *_smoothed_rsp folders.

```
biswebnode smoothimage -i divided_zscore/${div_type}/\$file -o \$tmpfile --sigma 3.0 --inmm true

biswebnode resampleimage -i \$tmpfile -o \${sm_folder}/\$file xsp 2.0 --ysp 2.0 --zsp 2.0

biswebnode concatenateimages -o \${sm_folder}/\${genotype} concatenation.nii.gz -i
\${sm_folder)/r_*.nii.gz
```

c. Next, *afni_roi.sh* runs the following steps using AFNI:
   i. 3D t-test (3dttest++): a 3D, voxel-by-voxel t-test is performed to establish voxels of fluorescence intensity that are statistically significant between experimental and control groups.

Note: afni_roi.sh performs the 3dttest++ calculation using the concatenated files (*_*concatenation.nii.gz*) in the *_smoothed_rsp folders.

   ii. Output: A residual file (*_*vs_wt_roi_resid_C0divC1.nii.gz*) representing the difference between the data values of the comparison group from the set mean, and a six-frame t-test file (*_*vs_wt_roi_ttest_C0divC1.nii.gz*) are generated. The six frames are as follows: **Frame0:** SetA-SetB_mean = difference of means; **Frame1:** SetA-SetB_Zscr; **Frame2:** SetA_mean = mean of SetA; **Frame3:** SetA_Zscr; **Frame4:** SetB_mean = mean of SetB; **Frame5:** SetB_Zscr.

```
3dttest++ -prefix

\${dttest_folder}/\${genotype}_vs_wt_roi_ttest_${div_type}.nii.gz

-resid

\${dttest_folder}/\${genotype}_vs_wt_roi_resid_${div_type} .nii.gz

-setA \${sm_folder}/\${genotype}_concatenation.nii.gz -setB

\${wt_folder}/wt_concatenation.nii.gz -toz
```

   iii. 3D full-width at half max (3DFWHMx): The spatial autocorrelation of the data is performed to estimate the approximate spatial noise distribution of the data.

Note: The 3dFWHMx command (3dFWHMx) uses the residual file (*_*vs_wt_roi_resid_C0divC1.nii.gz*) generated in the 3dttest step.

```
3dFWHMx -input

\${dttest_folder}/\${genotype}_vs_wt_roi_resid_${div_type}.nii.gz

-mask ${ZBRAIN_BINARY_FILE} -acf -detrend | (tee
\${dClustSim_folder}/\${genotype}.num >&2 )

cp 3dFWHMx.1D \${dFWHMx_folder}/\${genotype}_3dFWHMx.1D

cp 3dFWHMx.1D.png \${dFWHMx_folder}/\${genotype}_3dFWHMx.1D.png
```

iv. Output: The **3dFWHMx** step outputs a regression plot file (**_3dFWHMx.1D.png*) and three model parameters of the autocorrelation data in a separate file (**_3dFWHMx.1D*) for each comparison group (e.g., *hom_3dFWHMx.1D*, *het_3dFWHMx.1D*) in the subfolder, **3dFWHMx.**

v. **3D cluster simulation** (**3dClustSim**)**:** Based on the noise parameters determined by **3DFWHMx**, critical sizes of spatial clusters of intensity are determined. Depending on the statistical threshold used, any cluster of statistically significant intensity smaller than this critical size is deemed to be noise.

Note: ***afni_roi.sh*** applies the **3dClustSim** command (***3dClustSim***) and uses the three model parameters generated by the **3dFWHMx** step to estimate the probability of false positive clusters. The three model parameters are stored in the file, *genotype.num* and are saved in the **3dClustSim** subfolder.

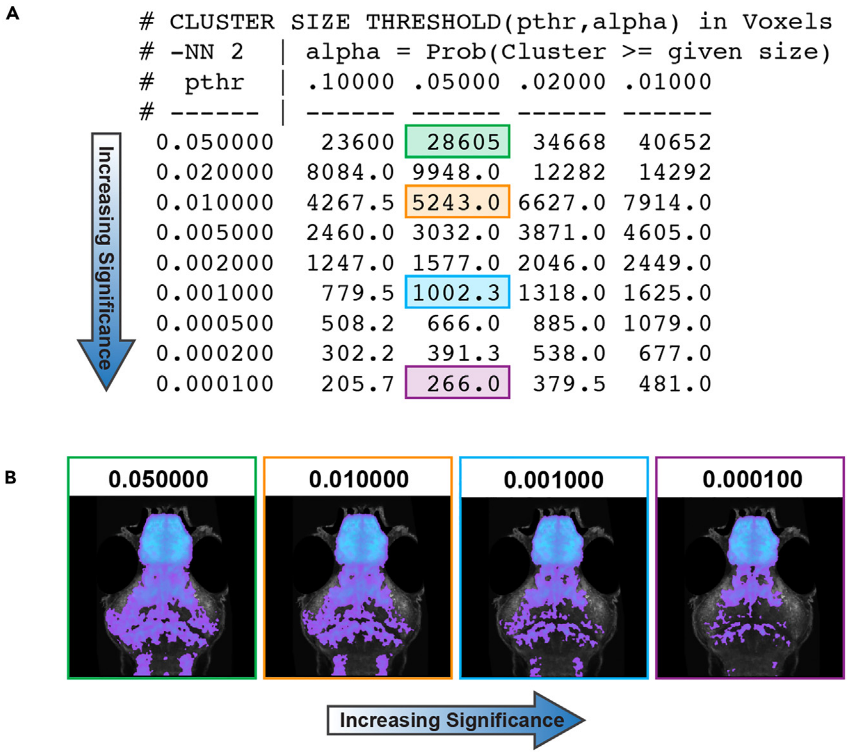
vi. The **3dClustSim** step outputs nine text files per comparison group (*.1D), each with different thresholding and clustering approaches in the subfolder, **3dClustSim.**

```
NUMBER=\$(tail -1 \${dClustSim_folder}/\${genotype}.num | awk

'{print \$1,\$2,\$3}')

3dClustSim -mask ${ZBRAIN BINARY_FILE} -acf \${NUMBER} -

prefix

\${dClustSim_folder}/\${genotype}_${div_type}_wt_roi_nomask

rm -f 3dFWHMx.1D 3dFWHMx.1D.png
```

vii. Output: The output files for ***afni_roi.sh*** will appear in the subfolders: **C0divC1** or **C2divC1.**

14. In preparation for visualizing differences between groups using AFNI, extract Frame 1 (SetA-SetB_Zscr) from the six-frame image generated by **3dttest++** in Step 13c. This frame contains the difference in z-score between the experimental and control groups. This is done locally using the BioImage Suite WebApp Viewer Tool (https://bioimagesuiteweb.github.io/webapp/viewer.html#).
    a. Load (File → Load Image) the standard reference fish (*standard_ref.nii.gz*) and the t-test file (**_vs_wt_roi_ttest_C0divC1.nii.gz*) generated in the **3dttest++** step as an overlay (File → Overlay → Load Overlay).
    b. In the "Image Processing" drop-down menu, select "Extract Frame."
    c. In the "Viewer Controls" menu, scroll to the "Extract Frame" tab.
    d. To extract Frame 1, select "Overlay" for both "Input" and "Output," type "1" in the "Frame" box, and select the "Execute" button. Do not edit the advanced settings.
    e. Save the Frame 1 file by selecting "Overlay" → "Save Overlay."
    f. Load the Frame 1 file (**roi_ttest_Frame_001_C0divC1.nii.gz*) into the **afni_roi** folder.

⚠ CRITICAL: The frames generated by 3dttest++ are zero-indexed. Therefore, the first frame is designated as Frame 0, and the second, as Frame 1. BioImage Suite accounts for zero-indexing, and entering 1 into the Frame option as described above will extract the appropriate frame.



```
# CLUSTER SIZE THRESHOLD(pthr,alpha) in Voxels
# -NN 2    | alpha = Prob(Cluster >= given size)
#  pthr    | .10000 .05000 .02000 .01000
# ------   | ------ ------ ------ ------
 0.050000    23600  28605  34668  40652
 0.020000   8084.0 9948.0  12282  14292
 0.010000   4267.5 5243.0 6627.0 7914.0
 0.005000   2460.0 3032.0 3871.0 4605.0
 0.002000   1247.0 1577.0 2046.0 2449.0
 0.001000    779.5 1002.3 1318.0 1625.0
 0.000500    508.2  666.0  885.0 1079.0
 0.000200    302.2  391.3  538.0  677.0
 0.000100    205.7  266.0  379.5  481.0
```

Figure 2. Representative brain activity images visualized at different statistical thresholds generated by cluster filtering in AFNI

(A) Example of an output table generated by AFNI (3dClustSim). The vertical axis represents a critical p-value for a voxel-by-voxel t-test, and the horizontal axis represents the critical p-value for a given noise cluster. Note that minimum permissible cluster sizes decrease with a more stringent critical p-value on the vertical axis (as more noise voxels are eliminated by the threshold), as shown in the colored boxes, but increase with a more stringent p-value on the horizontal axis (as larger clusters are required to be deemed non-noise).

(B) Brain activity map of *scn1lab*$^{\Delta 44/\Delta 44}$ mutant zebrafish shown at various critical p-values. Colors correspond to the boxes shown in (A).

Note: Each experimental group (homozygous and heterozygous) will have a Frame 1 file and should be labeled with the naming convention: *experimentalgroup_vs_wt_roi_ttest_Frame_001_C0divC1.nii.gz*. For example, "*het_vs_wt_roi_ttest_Frame_001_C0divC1.nii.gz.*"

Note: Checkpoint: In the **afni_roi** folder, there should be two subfolders (**C0divC1**, **C2divC1**) and the Frame 1 file.

15. To identify significant clusters for activity and cell-specific marker images, perform cluster filtering using the values generated by AFNI and the BioImage Suite WebApp Viewer Tool (https://bioimagesuiteweb.github.io/webapp/viewer.html#) locally.
    a. To perform cluster filtering on activity or cell-specific marker images, execute the bash script, ***clusterfilter.sh***. After uploading Frame 1 to the **afni_roi** folder, open the terminal and change directories to the **experiment folder** and type ***bash clusterfilter.sh*** in the terminal command line followed by the number designating the desired p-value threshold (e.g., ***bash clusterfilter.sh 0.05***, ***clusterfilter.sh 0.02***, ***clusterfilter.sh 0.0001***).
    b. After running the script, ***clusterfilter.sh***, the thresholded files (e.g., *ttest_* _C0divC1.nii.gz*) will appear in a subfolder called **thresholded** (e.g., afni_roi > C0divC1 > thresholded).

⚠ CRITICAL: To determine which statistical thresholds to use, we recommend visualizing images at multiple statistical thresholds generated by AFNI (Figure 2) and comparing these images to the results of the statistical analysis of regional differences in brain activity using linear mixed models (see quantification and statistical analysis). The p-values and cluster sizes for filtering the data will appear in one of the nine 3dClustSim tables generated, one for each type of simulation performed, based upon nearest neighbor clustering degree (face, edge, or vertex) and significance tests (two-tailed, one-tailed, or bi-sided). We recommend simulating cluster sizes using a second-degree nearest neighbor clustering with a bi-sided significance test (cluster sizes detailed in **_nomask.NN2_bisided.1D*).

Note: The thresholded files are named by the following convention: *ttest_expfolder_Frame001_roi_genotype_p-value_clustersize_cellspecificmarker.nii.gz*, where *expfolder* is the experiment folder; *Frame001* is the extracted frame 1; *genotype* is homozygous or heterozygous; *p-value* refers to the z-scored p-value (e.g., p-value of 0.05 corresponds to a z-score of 1.96); *clustersize* refers to the corresponding critical cluster size in the **.1D* file; *cellspecificmarker* refers to either *C0divC1* or *C2divC1*. For example, *"ttest_scn1lab_exp1_Frame001_roi_hom_1.96_23.7_C0divC1.nii.gz."*

⚠ CRITICAL: Because brain volume images are downsampled, cluster filtering may mask biologically relevant differences in brain size. Therefore, we recommend applying cluster filtering only to activity and cell-specific datasets. Alternatively, AFNI may be used to generate the Frame 1 (z-score) image for Jacobian images (*mask_ja*.nii.gz*), as described in Steps 16–17 below, which may be visualized directly. For this reason, the cluster filtering script described in the previous step is only used for visualizing brain activity datasets. Significant regional differences in brain volume may be determined using linear mixed modeling, as described in quantification and statistical analysis.

16. In a bash terminal, execute the script ***afni_vol.sh*** in the **experiment** folder. This will generate the statistical files for brain volume followed by a list of the experimental and control groups. In the command line, enter ***bash afni_vol.sh wt het hom***.
    a. Input:

```
PWD=$(basename ‘pwd‘)

BINDIR="/usr/local/bin"

ZBRAIN_BINARY_FILE="Zbrain_binary_mask_rsp.nii.gz"

GENO_TYPE="$@"

GENO_TYPE="${GENO_TYPE//[wW][tT]/}"
```

       i. BINDIR: Path of bin directory containing all files and scripts.
       ii. ZBRAIN_BINARY_FILE: *Zbrain_binary_mask_rsp.nii.gz*.
       iii. GENO_TYPE: Genotype (e.g., heterozygous, homozygous, and wild-type).

⚠ CRITICAL: The location of the bin folder must be specified in *afni_vol.sh*.

    b. ***afni_vol.sh*** uses the z-score files (*mask_ja*_zscore.nii.gz*) and runs the smoothing steps in Bioimage Suite. As these files are generated, they are placed in the subfolder, **afni_vol**, separated by genotype.

Note: Because the volume images are already downsampled, no further downsampling is required in this step (unlike in *afni_roi.sh*) to maintain the computational feasibility of the multiple-comparisons correction process.

c. *afni_vol.sh* performs the same smoothing, 3dttest, 3dFWHMx, and 3dClustSim steps as described above in Steps 13b–13c.

Note: The Jacobian map calculation is inherently smooth because it is computed off a smooth tensor spline non-linear transformation. Due to this smoothness, even large clusters of changes may be falsely deemed as noise because methods such as 3dClustSim assume that each voxel represents an independent measurement. This, while true for fMRI data (which is what AFNI was developed for), is an overly conservative assumption for Jacobian maps where adjacent voxels are highly correlated due to the inherent smoothness of the underlying transformation. Therefore, a 3-dimensional t-test alone may suffice for visualizing biologically relevant differences in brain volume.

```
biswebnode smoothimage -i jacobian_zscore/\$file -o

\${sm_folder}/\$ {file} --sigma 3.0 --inmm true

biswebnode concatenateimages -o

\${sm_folder}/\${genotype}_concatenation.nii.gz -i

\${sm_folder}/mask_ja_*.nii.gz
```

```
3dttest++ -prefix

\${dttest_folder}/\${genotype} vs_wt_volume_ttest.nii.gz -resid

\${dttest_folder}/\${genotype}_vs_wt_volume_resid.nii .gz - setA

\${sm_folder}/\${genotype} concatenation.nii.gz -setB

\${wt_folder} /wt_concatenation.nii.gz -toz
```

```
3dFWHMx -input

\${dttest_folder}/\${genotype} _vs_wt_volume_resid.nii.gz -mask

${ZBRAIN_BINARY_FILE} -acf -detrend | (tee

\${dClustSim_folder}/\${genotype}.num >&2 )

cp 3dFWHMx.1D \${dFWHMx_folder}/\${genotype} _3dFWHMx.1D

cp 3dFWHMx.1D.png \${dFWHMx_folder}/\${genotype} 3dFWHMx.1D.png
```

```
NUMBER=\$ (tail -1 \${dClustSim_folder}/\${genotype}.num | awk

'{print \$1,\$2,\$3}')

3dClustSim -mask ${ZBRAIN_BINARY FILE} -acf \${NUMBER} -prefix

\${dClustSim_folder}/\${genotype}_wt_vol_nomask

rm -f 3dFWHMx.1D 3dFWHMx.1D.png
```

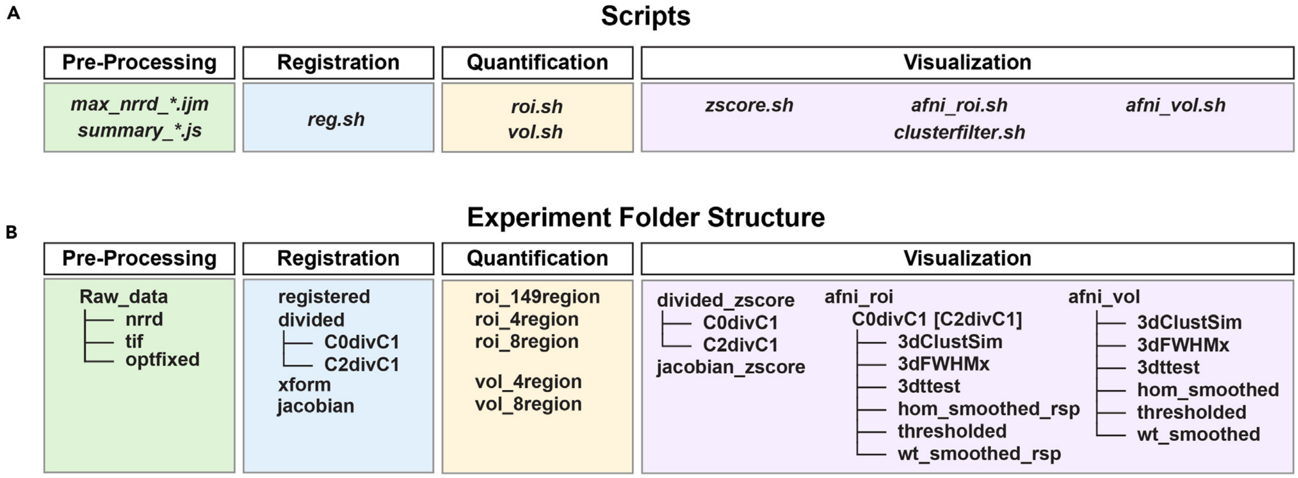


Figure 3. BREEZE-mapping scripts and output folder structure

(A) Scripts utilized at each major step in the pipeline.

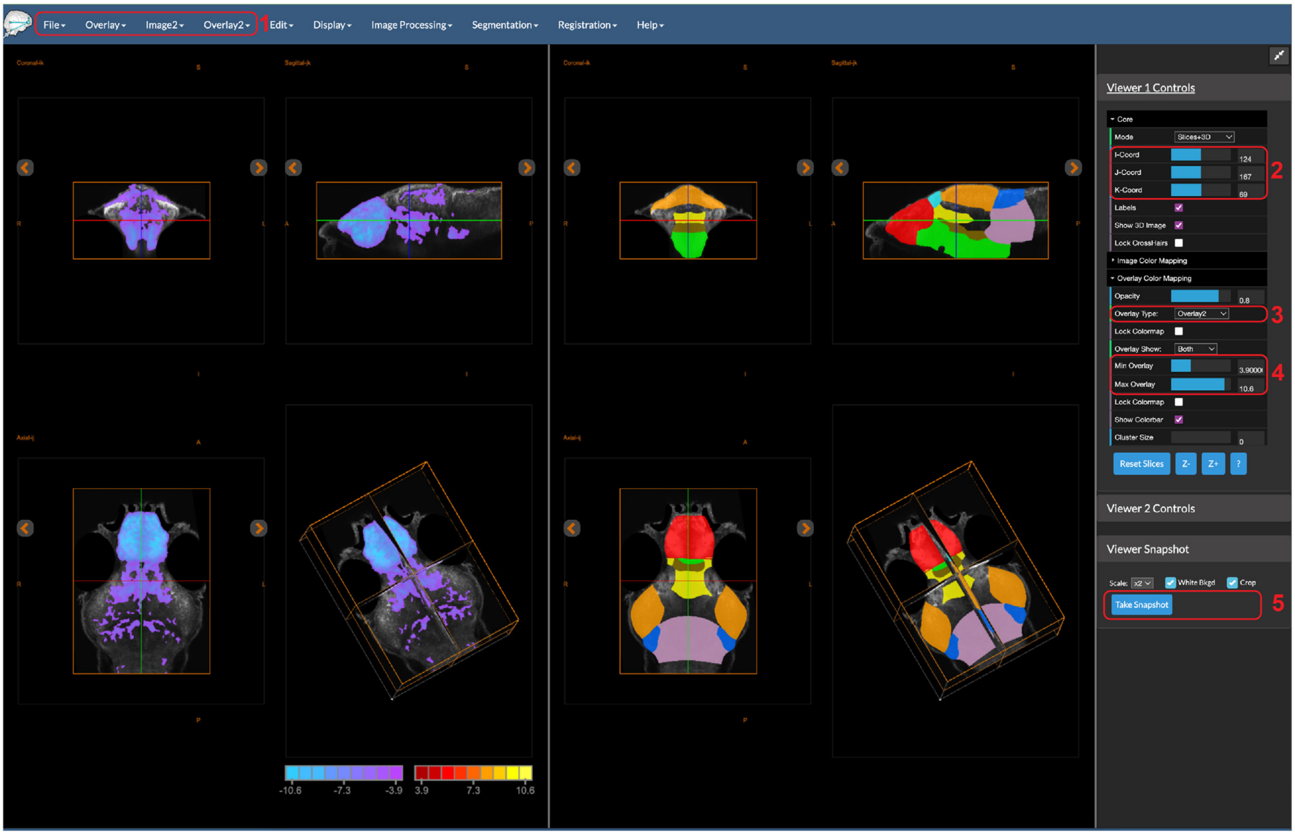
(B) Structure of output folders and subfolders generated at each major step.

d. Output: The output files for ***afni_vol.sh*** will appear in the folder, **afni_vol.**

17. To visualize volume differences between groups using AFNI, extract Frame 1 (SetA-SetB_Zscr) from the six-frame image generated by **3dttest++**. This frame contains the difference in z-score between the experimental and control groups. This is done locally using the BioImage Suite WebApp Viewer Tool, as described above for activity or cell-specific datasets in Step 14.

Note: Checkpoint: Confirm that all files and subfolders have been generated correctly by the BREEZE-mapping pipeline (Figure 3). The scripts used at each step are shown in Figure 3A and the corresponding file structure generated by the pipeline is shown in Figure 3B.

18. Use the BioImage Suite WebApp Overlay Viewer (https://bioimagesuiteweb.github.io/webapp/overlayviewer.html?load=https://bioimagesuiteweb.github.io/webapp/images/viewer.biswebstate) locally to visualize both activity/cell-specific marker cluster-filtered images and z-scored volume images.
    a. To visualize cluster-filtered brain activity or cell-specific marker images, download the folder, **thresholded**, in the **afni_roi** folder. Here, we describe how to visualize images using the Overlay Viewer.
        i. Using the Overlay viewer, select the "Orthogonal" tab and load the standard reference image (File → Load Image → *standard_ref_rsp.nii.gz*).
        ii. Next, load the Frame 1 image cluster-filtered at the appropriate threshold (e.g., Overlay → Load Overlay → **Frame_001_C0divC1.nii.gz.*)
        iii. To set the minimum (Min) and maximum (Max) values for visualization, use the z-score of the p-value threshold as the Min (e.g., p-value of 0.01 would have a minimum of 2.58). Users may select the Max based on characteristics of the specific dataset.
        iv. To change the color scheme, locate the "Viewer Control" and go to the drop-down menu, "Overlay Color Mapping." Select "Overlay2" for the "Overlay Type."
        v. The crosshairs can be used to navigate the zebrafish brain and locate changes in intensity slice-by-slice. To identify the exact coordinates, locate the 'I' 'J' 'K' coordinates in the menu control, which are equivalent to X, Y, Z.
    b. To visualize z-scored brain volume images (not cluster filtered), download the extracted Frame 1 files in the **afni_vol** folder. Z-scored volume images may be visualized using the Overlay Viewer, as described above, with some modifications.



Viewer 1 Controls

Core

Mode	Slices+3D
I-Coord	124
J-Coord	167
K-Coord	69
Labels	☑
Show 3D Image	☑
Lock CrossHairs	□

Image Color Mapping

Overlay Color Mapping

Opacity	0.8
Overlay Type:	Overlay2
Lock Colormap	□
Overlay Show:	Both
Min Overlay	3.9000
Max Overlay	10.6
Lock Colormap	□
Show Colorbar	☑
Cluster Size	0

Reset Slices Z- Z+ ?

Viewer 2 Controls

Viewer Snapshot

Scale: x2 ☑ White Bkgd ☑ Crop

Take Snapshot

Figure 4. Representative BioImage Suite Web Dual Viewer window

The BioImage Suite Web Dual Viewer provides the ability to compare two images side by side. The *scn1lab*$^{\Delta 44/\Delta 44}$ brain activity phenotype (left) is shown alongside the eight-region zebrafish brain atlas (right) in "Slices+3D" mode, which displays images in axial, coronal, sagittal, and 3D orientations. (#1) A standard reference image is loaded initially as an image (File → Load Image; Image2 → Load Image), after which brain images and atlases may be viewed as overlays (Overlay → Load Overlay; Overlay2 → Load Overlay). (#2) Individual positions may be specified along the x, y, and z axes (i.e., i., j., and k), which will be reflected in both images. (#3) Color mapping for images may be specified with the Overlay Type parameter. Images in this protocol are typically visualized with the "Overlay2" color mapping. (#4) Color map thresholding may be performed with the "Min Overlay" (intensity value below which a voxel is not shown) and "Max Overlay" (intensity value above which a maximum saturation will be shown) options. (#5) A snapshot of both viewers may be taken with the "Take Snapshot" option.

i. Select the "Orthogonal" tab and load the standard resampled reference image (File → Load Image → *standard_ref_rsp.nii.gz*).
ii. Next, load the Frame 1 image at the appropriate threshold (e.g., Overlay → Load Overlay → **volume_ttest_Frame_001.nii.gz*).
iii. Users may select the Min and Max z-score values for visualization based on characteristics of the specific dataset.

c. Another viewing tool available in BioImage Suite is the Dual Viewer (https://bioimagesuiteweb.github.io/webapp/dualviewer.html), which allows two images to be viewed side-by-side, or for an image to be viewed alongside the zebrafish brain atlas. Figure 4 shows the brain activity phenotype in *scn1lab*$^{\Delta 44/\Delta 44}$ mutants viewed alongside the 8-region atlas using the BioImage Suite Dual Viewer. The Dual Viewer may also be used to visualize brain volume images.
   i. In "Viewer 1," load the standard reference image (File → Load Image → *standard_ref_rsp.nii.gz).*
   ii. Next, in "Viewer 1," load the Frame 1 image as an overlay (Overlay → Load Overlay → **Frame_001_C0divC1.nii.gz* or **volume_ttest_Frame_001.nii.gz*).

iii. Set the Min and Max values as described above for the Overlay/Orthogonal Viewer (see Step 18a). For activity or cell-specific marker thresholded images, use the z-score of the p-value threshold as the minimum (Min) (e.g., p = 0.01 would have a minimum of 2.58) and maximum (Max) based on strictest p-value threshold limit. For volume Frame 1 (z-score images), the minimum and maximum can be determined by additional statistical testing performed on the raw "vol" data generated in Step 11c.

iv. In "Viewer 2," load the standard reference image (File → Load Image → *standard_ref_rsp.nii.gz*).

v. Next, in "Viewer 2," load the atlas file (e.g., *Zbrain_atlas_149region.nii.gz*) as an overlay (File → Load Overlay2 → *Zbrain_atlas_*region.nii,gz*).

vi. To visualize the atlas in "Viewer Control 2," go to the drop-down menu, "Overlay Color Mapping" and select "Objectmap" for the "Overlay Type." No Min/Max adjustments are required in "Viewer Control 2."

vii. The cross hairs will be in sync for both viewers, giving users the ability to hover over and navigate regions of image intensity changes.

d. Another viewing tool available in BioImage Suite is the Mosaic Viewer (same URL as Overlay Viewer), which allows users to visualize the data as a collection of z-slices.

i. To visualize images using the Mosaic Viewer, select the "Mosaic" tab.

ii. The "Viewer Control" tab is used to change parameters to visualize specific brain slices.

iii. The "Rows" and "Columns" can be used to select a specific number of slices that best displays the phenotype.

iv. The option "First" indicates the starting slice and "Increment" is the difference in slices between the first and second slices.

EXPECTED OUTCOMES

The BREEZE-mapping pipeline allows users to register larval zebrafish whole-brain images to a standard zebrafish reference brain, quantify regional differences in brain volume, activity, or other cell-specific markers across experimental groups using zebrafish brain atlases, and visualize brain phenotypes using the BioImage Suite WebApp Viewer Tool. The expected results of each of the major steps of the pipeline are: *(1)* ***Pre-processing:*** pre-processed (*.nii*) whole-brain z-stack images that are optimized for registration; *(2)* ***Registration:*** registered and resliced whole-brain images that are normalized to a uniform marker (tERK) and processed for volume quantification; *(3)* ***Quantification:*** z-scored images comparing brain volume or cell-specific marker phenotypes between experimental and control groups, and *.csv* files containing Jacobian values and normalized cell-specific marker values for regions of interest designated by a zebrafish brain atlas; *(4)* ***Visualization:*** brain volume and activity 3D maps for visualizing phenotypes, including cluster filtered maps displaying significant clusters in the experimental versus control groups.

The BREEZE-mapping pipeline is expected to generate files that allow for the visualization and quantification of brain volume and/or activity phenotypes in zebrafish mutants of a gene of interest. Figure 5 highlights representative results from Weinschutz Mendes et al. (2023)[1] showing how our pipeline may be used to visualize 3D brain volume and/or activity phenotypes in zebrafish mutants of ASD-associated genes and to quantify differences in volume and/or activity in regions of interest designated by zebrafish brain atlases. Figures 5A–5D show the visualization of a robust brain volume phenotype in *dyrk1aa*$^{\Delta 77/\Delta 77}$*dyrk1ab*$^{\Delta 8/\Delta 8}$ mutants (Figure 5A), which carry frameshift mutations in the zebrafish orthologs of the ASD gene, *DYRK1A*, in sequential slices using the Mosaic Viewer in BioImage Suite (Figure 5B), and quantification of significantly reduced brain volume in the forebrain (telencephalon) of mutants using the 8-region zebrafish brain atlas (Figures 5C and 5D). Voxel-wise Z-score values representing brain volume differences are shown. Figures 5E–5H shows the visualization of a robust brain activity phenotype in *scn1lab*$^{\Delta 44/\Delta 44}$ mutants (Figure 5E), which carry frameshift mutations in a zebrafish ortholog of the ASD genes, *SCN1A* and *SCN2A*, using the Mosaic Viewer in BioImage Suite (Figure 5F), and quantification of significantly decreased brain activity in one region, the pretectal dopaminergic cluster, using the 149-region zebrafish brain atlas (Figures 5G and 5H).

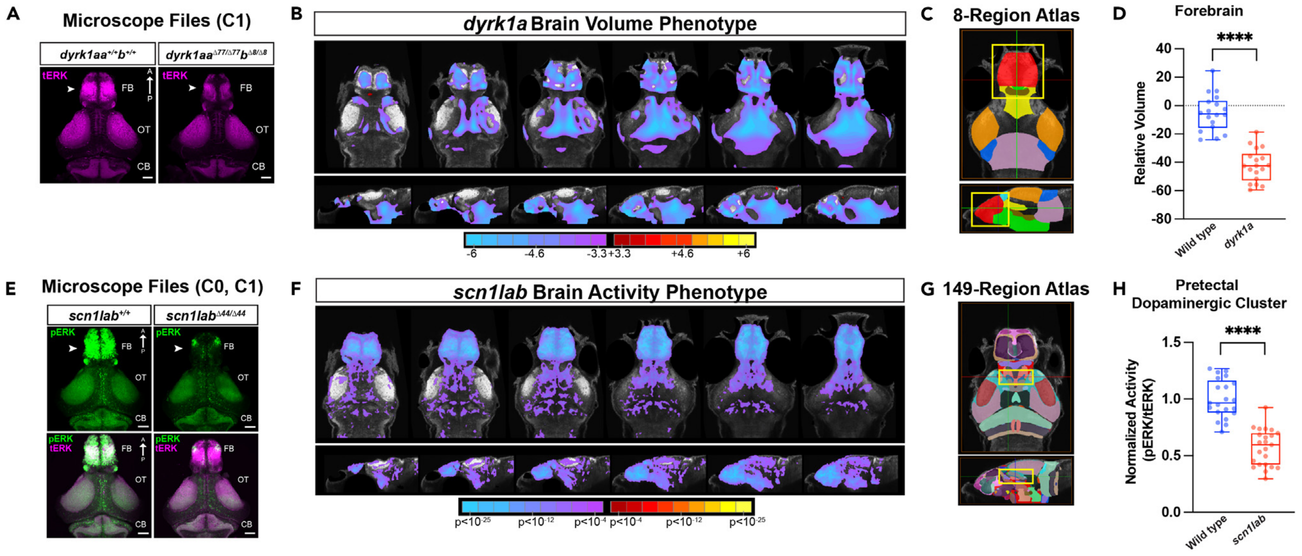


Figure 5. Representative quantification of regional brain volume and activity phenotypes in zebrafish mutants of ASD risk genes

(A) tERK-immunostained brains of *dyrk1aa*$^{\Delta 77/\Delta 77}$*dyrk1ab*$^{\Delta 8/\Delta 8}$ and wild-type larvae at 6 dpf. Note the reduction in forebrain volume in mutants (arrowheads). Dorsal views. FB, forebrain; OT, optic tectum; CB, cerebellum. Scale bar = 50 μm.

(B) Voxel-wise Z-score values representing brain volume differences in *dyrk1aa*$^{\Delta 77/\Delta 77}$*dyrk1ab*$^{\Delta 8/\Delta 8}$ versus background-matched wild-type larvae. Images shown from left to right represent sequential slices: axial views, top row, dorsal to ventral; sagittal views, bottom row, lateral to medial. Scale bar represents Z-score (red/yellow, increased in mutant; cyan/purple, decreased in mutant).

(C) Eight-region zebrafish brain atlas derived from Thyme et al. (2019)[17] and Randlett et al. (2015).[16] Yellow box (region shown in red), forebrain.

(D) Regional differences in forebrain volume of *dyrk1aa*$^{\Delta 77/\Delta 77}$*dyrk1ab*$^{\Delta 8/\Delta 8}$ (n = 18) and wild-type (n = 18) relative to the standard zebrafish reference brain[16] (dotted line). ****$p < 0.0001$, (one-way ANOVA).

(E) pERK- (top) and merged pERK- and tERK-immunostained brains of *scn1lab*$^{\Delta 44/\Delta 44}$ and wild-type larvae at 6 dpf. Note the decrease in pERK staining in the mutant forebrain (arrowheads). Dorsal views. FB, forebrain; OT, optic tectum; CB, cerebellum. Scale bar = 50 μm.

(F) Voxel-wise Z-score pERK/tERK values representing brain activity differences in *scn1lab*$^{\Delta 44/\Delta 44}$ versus background-matched wild-type larvae. Images shown from left to right represent sequential slices: axial views, top row, dorsal to ventral; sagittal views, bottom row, lateral to medial. Scale bar represents Z-score (red/yellow, increased in mutant; cyan/purple, decreased in mutant). Results are shown at $p < 0.05$ whole-brain family-wise error (FWE) corrected with an initial *P* threshold of 0.0001.

(G) 149-region zebrafish brain atlas derived from Randlett et al. (2015).[16] Yellow box and crosshairs indicate location of the pretectal dopaminergic cluster.

(H) Normalized brain activity (pERK/tERK) in *scn1lab*$^{\Delta 44/\Delta 44}$ versus background-matched wild-type fish in the pretectal dopaminergic cluster. ****$p < 0.0001$ (one-way ANOVA).

Adapted from Weinschutz Mendes et al. (2023).[1]

The activity phenotype results are displayed at $p < 0.05$ whole-brain family-wise error (FWE) corrected with an initial *P* threshold of 0.0001 using AFNI cluster filtering.

To highlight the breadth of the BREEZE-mapping pipeline, Figure 6 shows hierarchical clustering of baseline brain activity fingerprints in zebrafish mutants of ten ASD-associated genes using the 149-region atlas, as shown in Weinschutz Mendes et al. (2023).[1] We found that these fingerprints broadly cluster by major brain subdivision (telencephalon, diencephalon, mesencephalon, and rhombencephalon) using the four-region atlas (Figure 6A). Figure 6B shows 131 brain regions with significant differences in baseline activity in at least two mutants by linear mixed models.[1] This demonstrates how the pipeline may be used to examine brain activity phenotypes across multiple zebrafish mutant lines. Taken together, BREEZE-mapping represents a powerful tool to identify quantifiable whole-brain activity and/or size phenotypes in larval zebrafish and provides a user-friendly, interactive website for visualizing these phenotypes.

QUANTIFICATION AND STATISTICAL ANALYSIS

To identify significant differences in regional brain activity and volume between experimental and control groups, we utilized linear mixed models (LMM).[1] We recommend applying LMM to analyze the *roi.csv* and

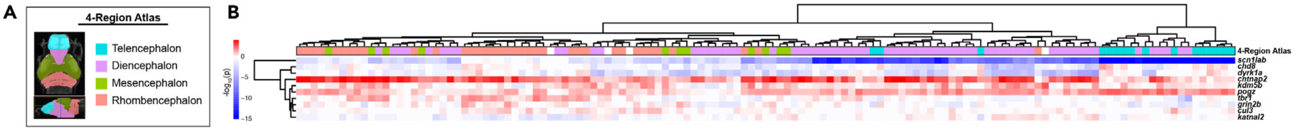


Figure 6. BREEZE-mapping analysis of baseline brain activity in zebrafish mutants of 10 ASD risk genes

(A) Four-region atlas derived from Randlett et al. (2015)[16] shows the four major brain subdivisions: telencephalon, diencephalon, mesencephalon, rhombencephalon.

(B) Hierarchical clustering of 131 regions with significant differences in baseline brain activity in at least two mutants by linear mixed models using the 149-region atlas derived from Randlett et al. (2015).[16] Brain activity fingerprints are shown for the following mutants: *scn1lab*$^{\Delta 44/\Delta 44}$; *chd8*$^{\Delta 5/\Delta 5}$; *dyrk1aa*$^{\Delta 77/\Delta 77}$*dyrk1ab*$^{\Delta 8/\Delta 8}$; *cntnap2a*$^{\Delta 121/\Delta 121}$*cntnap2b*$^{31ins/31ins}$; *kdm5ba*$^{\Delta 17/\Delta 17}$*kdm5bb*$^{\Delta 14/\Delta 14}$; *pogza*$^{\Delta 23/\Delta 23}$*pogzb*$^{\Delta 20/\Delta 20}$; *tbr1a*$^{\Delta 64/\Delta 64}$*tbr1b*$^{\Delta 10/\Delta 10}$; *grin2ba*$^{\Delta 25/\Delta 25}$*grin2bb*$^{\Delta 64/\Delta 64}$; *cul3a*$^{\Delta 7/\ \Delta 7}$*cul3b*$^{\Delta 20/+}$; *katnal2* $^{\Delta 4/\Delta 4}$. The scale bar represents the signed -log$_{10}$-transformed p-values from linear mixed models.

Adapted from Weinschutz Mendes et al. (2023).[1]

vol.csv datasets, because this method accounts for variability associated with the day of testing. Specifically, we account for variations of brain activity (pERK/tERK) or volume by including the date of the experiment as a random effect in LMM. Regional differences in activity or volume across different mutants may be visualized using hierarchical clustering analysis based on the signed -log$_{10}$-transformed p-values from LMM, where sign is the direction of the difference in activity or volume when comparing two conditions. Clustergram (“pheatmap: Pretty heatmaps” [RRID:SCR_016418]: https://cran.r-project.org/web/packages/pheatmap/index.html) may be used for visualization of brain region clusters in a heatmap.

1. Assess the differences in regional brain activity and volume between experimental and control groups.
   a. Import data from multiple sheets in *roi.xlsx* or *voi.xlsx*: each sheet records the data for one mutant on one experiment date, in which rows are sample names indicating their group (homozygous, heterozygous, or wild-type), and columns are measures within brain regions.
2. Apply LMM to compare: (1) homozygous versus wild-type; and (2) heterozygous versus wild-type, to obtain regional brain activity and volume differences and their associated p-values.

```
res <- lme4::lmer(y~grps + (1|exp), data = dat)

beta = summary(res)$"coefficients"[2, 1]

Pval = anova(res)$’Pr(>F)’[1]
```

Note: For each gene, a data frame *dat* is created to combine all experimental data on samples in groups of interest from different dates (e.g., homozygous and wild-type, heterozygous and wild-type). The experimental date of samples is saved in column *exp*, and the group label of samples is saved in column *grps*.

3. Visualize mutant and brain region clusters based on the signed -log$_{10}$-transformed p-values.

```
score <- sign(beta) * -log10(Pval)

pheatmap::pheatmap(score_mat)
```

Note: The signed -log$_{10}$-transformed p-values are saved in *score*. This score for all genes is saved in a matrix, *score_mat*, where rows are genes and columns are brain regions.

LIMITATIONS

Similar to other registration methods, BREEZE-mapping relies on using a standard zebrafish reference brain and brain atlases,[16] which at present are available for larvae at 6 days post fertilization (dpf). For this reason, our pipeline is currently limited by the availability of a standard reference brain and atlases at other developmental stages. However, once atlases are developed for other stages, they can be readily utilized by our pipeline. Our method also registers brains to the standard zebrafish reference brain,[16] which uses tERK as a uniform label. To ensure optimal registration, it is important to use tERK-stained brains. However, it is possible to select images of brains with other neuronal labels as an alternative reference. In addition, because brains are mounted with the dorsal side facing the imaging objective, the resolution of the dorsal images in the z-stacks is often superior and image quality diminishes to some extent in the ventral regions.

Due to the computational resources required, it is recommended that the registration step is performed using an HPCC, particularly for larger datasets, which may limit pipeline accessibility. Another limitation is that the BREEZE-mapping scripts are optimized for experiments analyzing phenotypes in genetic mutants (i.e., comparing phenotypes in homozygous and heterozygous mutants to a wild-type control group), and the script nomenclature is written accordingly. For studies using different comparison groups (e.g., drug treatment, environmental exposures), editing the nomenclature in the scripts is required.

TROUBLESHOOTING

Problem 1

The ***max_nrrd_2.ijm*** or ***max_nrrd_3.ijm*** scripts are not processing the images (see "pre-processing," Step 2).

Potential solution

- Verify that the naming convention follows all the proper guidelines (e.g., **_C1.nii.gz, *_optfixed.nii.gz*). Individual naming conventions may be found at the respective steps.

Problem 2

The image orientation does not match the standard reference image (see "pre-processing," Step 2).

Potential solution

- Verify that the image was acquired in the dorsal to ventral direction.
- Reorient the image in ImageJ or BioImage Suite.

Problem 3

The script ***reg.sh*** does not output registered files (see "registration," Step 5).

Potential solution

- Verify that the pre-processed images have been uploaded to a folder named **optfixed** on the HPCC within a dedicated folder for each experiment.
- Change directories to the dedicated **experiment** folder, and run ***reg.sh*** from that folder.
- Re-run ***reg.sh*** after correcting these steps.

Problem 4

The orientation of the atlas does not match that of the image (see "quantification and statistical analysis," Step 9).

Potential solution

- Use the ***fslhd*** command of FSL to determine the orientation of the image. The header parameters qform_xorient, qform_yorient, qform_zorient, sform_xorient, sform_yorient, and sform_zorient should match between the atlas image and the image in question.

- Verify dimensionality of the atlas. The brain volume calculations use a downsampled atlas (i.e., voxel sizes are larger compared to the image intensity calculations). Usage of the downsampled atlas for calculation of image intensity (i.e., activity) will result in dimensionality errors.

Problem 5

The script ***zscore.sh*** does not output z-scored values and/or produces an error message (see "visualization," Step 12).

Potential solution

- Verify that the command ***bash zscore.sh*** is being run from the **experiment** folder. If not, change directories to the **experiment** folder and re-run the command.
- Verify that the command was followed by the experimental conditions (e.g., control, genotype, etc., in the format ***bash zscore.sh wt het hom*** to z-score three appropriately named experimental conditions).
- Lastly, verify that the naming convention of the genotypes remains consistent with the previous steps.
- Re-run ***zscore.sh*** after correcting these steps.

Problem 6

The scripts ***afni_vol.sh*** and ***afni_roi.sh*** run the smoothing and resampling steps, but the job fails immediately afterward (see "visualization," Steps 13 and 16).

Potential solution

- Verify that AFNI is properly installed onto the HPCC from which the script is being run.
- Verify that the ***afni_vol.sh*** and ***afni_roi.sh*** scripts are being run in the dedicated **experiment** folder.
- Verify that the ***afni_vol.sh*** command was followed by the experimental conditions (e.g., control, genotype, etc., in the format ***bash afni_vol.sh wt het hom*** for three conditions to be compared).
- Verify that the ***afni_roi.sh*** command was followed by the experimental conditions (e.g., control, genotype, etc., in the format ***bash afni_roi.sh wt het hom*** for three conditions to be compared).
- Re-run ***afni_vol.sh*** and/or ***afni_roi.sh*** after correcting these steps.

Problem 7

The script ***clusterfilter.sh*** does not output values and/or produces an error message (see "visualization," Step 15).

Potential solution

- Verify that the extracted activity frame is named ***Frame_001.nii.gz***.
- Verify that the extracted activity frame has been added to a folder named **afni_roi.**
- Verify that the command is being run from the **afni_roi** folder.
- Re-run ***clusterfilter.sh*** after correcting these steps.

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources and reagents should be directed to and will be fulfilled by the Lead Contact, Ellen J. Hoffman (ellen.hoffman@yale.edu).

Materials availability

The zebrafish mutant lines reported in this study are available upon request with a completed Materials Transfer Agreement.

Data and code availability

For the custom BREEZE-mapping codes reported in this study, go to GitHub: https://github.com/ehoffmanlab/Jin_Neelakantan_BREEZE_Mapping; Zenodo: https://doi.org/10.5281/zenodo.7996403. Sample datasets of whole-brain images from one experiment of zebrafish mutants of two ASD genes reported in Weinschutz Mendes et al. (2023)[1] are provided for users to test the codes. Original data have been deposited to Zenodo: https://doi.org/10.5281/zenodo.7996403.

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.xpro.2023.102647.

ACKNOWLEDGMENTS

We thank M. Carlson, S.E. Fitzpatrick, P. Jamadagni, A. Pruitt, and H. Weinschutz Mendes for discussions and H. Neelakantan for support in software development. This work was supported by the National Institutes of Health R01MH116002 and RM1MH132648, Binational Science Foundation (#2017325), Kavli Institute for Neuroscience at Yale University (Kavli Innovative Research Award), National Genetics Foundation, Simons Foundation (#573508, #345993), Spector Fund, and Swebilius Foundation (E.J.H.) and Interdepartmental Neuroscience Program at Yale (D.S.J.).

AUTHOR CONTRIBUTIONS

D.S.J., U.N., T.C., C.M.L., B.R., X.P., and E.J.H. developed and implemented the BREEZE-mapping pipeline. D.S.J. and U.N. wrote the scripts with input from C.M.L., X.P., and E.J.H. Y.L., W.W., and Z.W. developed and implemented LMM and statistical analysis of brain mapping datasets. D.S.J., U.N., C.M.L., Y.L., Z.W., X.P., and E.J.H. wrote the manuscript.

DECLARATION OF INTERESTS

X.P. is a consultant for the Brain Electrophysiology Laboratory Company. X.P. also consults and has an ownership stake in Veridat.

REFERENCES

1. Weinschutz Mendes, H., Neelakantan, U., Liu, Y., Fitzpatrick, S.E., Chen, T., Wu, W., Pruitt, A., Jin, D.S., Jamadagni, P., Carlson, M., et al. (2023). High-throughput functional analysis of autism genes in zebrafish identifies convergence in dopaminergic and neuroimmune pathways. Cell Rep. *42*, 112243. https://doi.org/10.1016/j.celrep.2023.112243.
2. McCammon, J.M., and Sive, H. (2015). Challenges in understanding psychiatric disorders and developing therapeutics: a role for zebrafish. Dis. Model. Mech. *8*, 647–656. https://doi.org/10.1242/dmm.019620.
3. Sakai, C., Ijaz, S., and Hoffman, E.J. (2018). Zebrafish Models of Neurodevelopmental Disorders: Past, Present, and Future. Front. Mol. Neurosci. *11*, 294. https://doi.org/10.3389/fnmol.2018.00294.
4. Shams, S., Rihel, J., Ortiz, J.G., and Gerlai, R. (2018). The zebrafish as a promising tool for modeling human brain disorders: A review based upon an IBNS Symposium. Neurosci. Biobehav. Rev. *85*, 176–190. https://doi.org/10.1016/j.neubiorev.2017.09.002.
5. Gerlai, R. (2023). Zebrafish (Danio rerio): A newcomer with great promise in behavioral neuroscience. Neurosci. Biobehav. Rev. *144*, 104978. https://doi.org/10.1016/j.neubiorev.2022.104978.
6. Kozol, R.A., Abrams, A.J., James, D.M., Buglo, E., Yan, Q., and Dallman, J.E. (2016). Function Over Form: Modeling Groups of Inherited Neurological Conditions in Zebrafish. Front. Mol. Neurosci. *9*, 55. https://doi.org/10.3389/fnmol.2016.00055.
7. Moreno-Mateos, M.A., Vejnar, C.E., Beaudoin, J.D., Fernandez, J.P., Mis, E.K., Khokha, M.K., and Giraldez, A.J. (2015). CRISPRscan: designing highly efficient sgRNAs for CRISPR-Cas9 targeting *in vivo*. Nat. Methods *12*, 982–988. https://doi.org/10.1038/nmeth.3543.
8. Kroll, F., Powell, G.T., Ghosh, M., Gestri, G., Antinucci, P., Hearn, T.J., Tunbak, H., Lim, S., Dennis, H.W., Fernandez, J.M., et al. (2021). A simple and effective F0 knockout method for rapid screening of behaviour and other complex phenotypes. Elife *10*, e59683. https://doi.org/10.7554/eLife.59683.
9. Hwang, W.Y., Fu, Y., Reyon, D., Maeder, M.L., Tsai, S.Q., Sander, J.D., Peterson, R.T., Yeh, J.R.J., and Joung, J.K. (2013). Efficient genome editing in zebrafish using a CRISPR-Cas system. Nat. Biotechnol. *31*, 227–229. https://doi.org/10.1038/nbt.2501.
10. Hwang, W.Y., Fu, Y., Reyon, D., Maeder, M.L., Kaini, P., Sander, J.D., Joung, J.K., Peterson, R.T., and Yeh, J.R.J. (2013). Heritable and precise zebrafish genome editing using a CRISPR-Cas system. PLoS One *8*, e68708. https://doi.org/10.1371/journal.pone.0068708.
11. Burgess, H.A., and Granato, M. (2007). Modulation of locomotor activity in larval zebrafish during light adaptation. J. Exp. Biol. *210*, 2526–2539. https://doi.org/10.1242/jeb.003939.
12. Burgess, H.A., and Granato, M. (2007). Sensorimotor gating in larval zebrafish. J. Neurosci. *27*, 4984–4994. https://doi.org/10.1523/JNEUROSCI.0615-07.2007.
13. Schoonheim, P.J., Arrenberg, A.B., Del Bene, F., and Baier, H. (2010). Optogenetic localization and genetic perturbation of saccade-generating neurons in zebrafish. J. Neurosci. *30*, 7111–7120. https://doi.org/10.1523/JNEUROSCI.5193-09.2010.
14. Guo, S. (2009). Using zebrafish to assess the impact of drugs on neural development and function. Expert Opin. Drug Discov. *4*, 715–726. https://doi.org/10.1517/17460440902988464.
15. Lovett-Barron, M., Andalman, A.S., Allen, W.E., Vesuna, S., Kauvar, I., Burns, V.M., and Deisseroth, K. (2017). Ancestral Circuits for the Coordinated Modulation of Brain State. Cell *171*, 1411–1423.e17. https://doi.org/10.1016/j.cell.2017.10.021.

16. Randlett, O., Wee, C.L., Naumann, E.A., Nnaemeka, O., Schoppik, D., Fitzgerald, J.E., Portugues, R., Lacoste, A.M.B., Riegler, C., Engert, F., and Schier, A.F. (2015). Whole-brain activity mapping onto a zebrafish brain atlas. Nat. Methods *12*, 1039–1046. https://doi.org/10.1038/nmeth.3581.

17. Thyme, S.B., Pieper, L.M., Li, E.H., Pandey, S., Wang, Y., Morris, N.S., Sha, C., Choi, J.W., Herrera, K.J., Soucy, E.R., et al. (2019). Phenotypic Landscape of Schizophrenia-Associated Genes Defines Candidates and Their Shared Functions. Cell *177*, 478–491.e20. https://doi.org/10.1016/j.cell.2019.01.048.

18. BioImage Suite Web. https://bioimagesuiteweb.github.io/webapp/.

19. Hoffman, E.J., Turner, K.J., Fernandez, J.M., Cifuentes, D., Ghosh, M., Ijaz, S., Jain, R.A., Kubo, F., Bill, B.R., Baier, H., et al. (2016). Estrogens Suppress a Behavioral Phenotype in Zebrafish Mutants of the Autism Risk Gene, CNTNAP2. Neuron *89*, 725–733. https://doi.org/10.1016/j.neuron.2015.12.039.

20. Cox, R.W. (1996). AFNI: software for analysis and visualization of functional magnetic resonance neuroimages. Comput. Biomed. Res. *29*, 162–173. https://doi.org/10.1006/cbmr.1996.0014.

21. Schindelin, J., Arganda-Carreras, I., Frise, E., Kaynig, V., Longair, M., Pietzsch, T., Preibisch, S., Rueden, C., Saalfeld, S., Schmid, B., et al. (2012). Fiji: an open-source platform for biological-image analysis. Nat. Methods *9*, 676–682. https://doi.org/10.1038/nmeth.2019.

22. Papademetris, X.,J.A., Schultz, R., Staib, L., Duncan, J., Barillot, C., Haynor, D., and Hellier, P. (2004). Integrated Intensity and Point-Feature Nonrigid Registration. In Medical Image Computing and Computer-Assisted Intervention—MICCAI 2004 (Springer), pp. 763–770.

23. Staib, L.H., Jackowski, M., and Papademetris, X. (2006). Brain Shape Characterization from Deformation. Proc. IEEE Int. Symp. Biomed. Imaging *3*, 1140–1143.

24. Najt, P., Wang, F., Spencer, L., Johnston, J.A.Y., Cox Lippard, E.T., Pittman, B.P., Lacadie, C., Staib, L.H., Papademetris, X., and Blumberg, H.P. (2016). Anterior Cortical Development During Adolescence in Bipolar Disorder. Biol. Psychiatry *79*, 303–310. https://doi.org/10.1016/j.biopsych.2015.03.026.