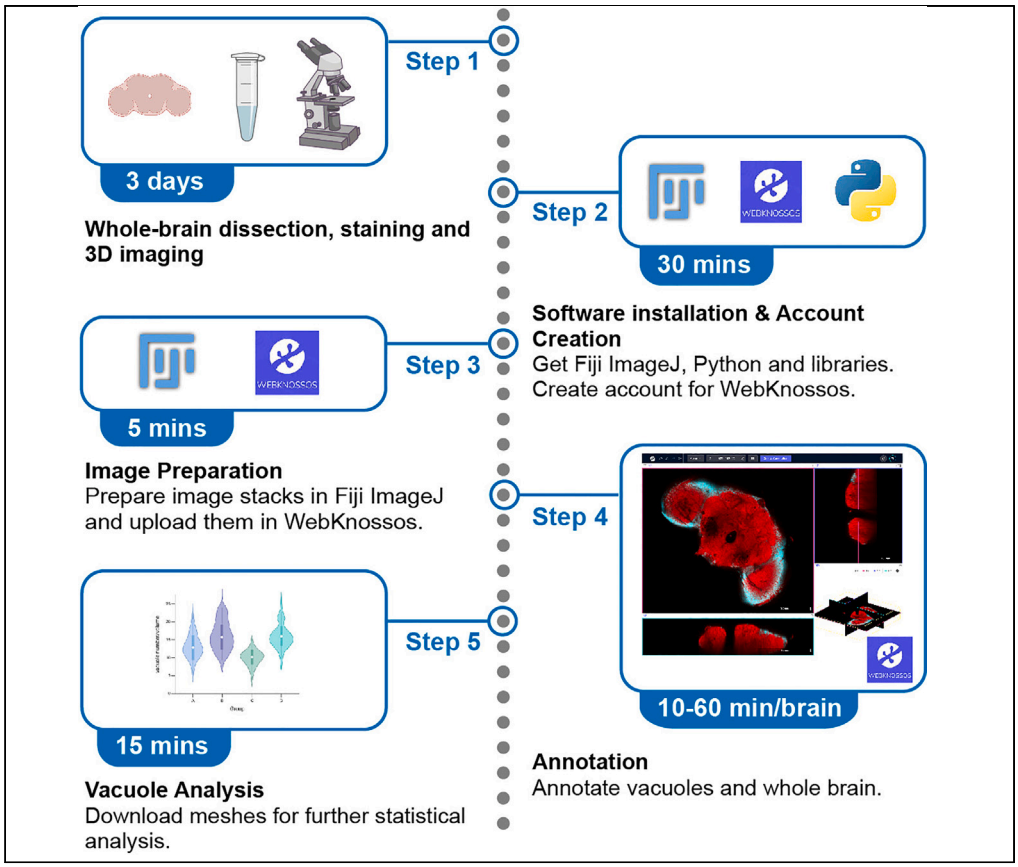
STAR Protocols

CellPress
OPEN ACCESS

Protocol

Protocol to analyze 3D neurodegenerative vacuoles in *Drosophila melanogaster*



Vacuole formation is a key hallmark of age-dependent neurodegeneration in the *Drosophila* brain. Here, we present a protocol to analyze 3D neurodegenerative vacuoles in the whole-mount *Drosophila melanogaster* brain. We describe steps for whole-brain dissection, staining, 3D imaging, and z-stack image processing using Fiji ImageJ. We then detail procedures for annotating and 3D-reconstructing neurodegenerative vacuoles with WEBKNOSSOS and Python, and performing statistical analysis in Python. This protocol enables measurement of parameters such as the number and volume of each vacuole.

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

Guangmei Liu, Shruthi Bandyadka, Kimberly McCall

kmccall@bu.edu

Highlights

A computational pipeline to perform vacuole volumetry in *Drosophila melanogaster* brain

Steps using WEBKNOSSOS to annotate neurodegenerative vacuoles in whole fly brain

Steps for statistical analysis and visualization of vacuole data in Python

Liu et al., STAR Protocols 5, 103017
June 21, 2024
https://doi.org/10.1016/j.xpro.2024.103017

STAR Protocols

CellPress
OPEN ACCESS

Protocol

Protocol to analyze 3D neurodegenerative vacuoles in *Drosophila melanogaster*

Guangmei Liu,[1,2,3] Shruthi Bandyadka,[1,2,3] and Kimberly McCall[1,4,*]

[1]Boston University, Boston, MA 02215, USA

[2]These authors contributed equally

[3]Technical contact

[4]Lead contact

*Correspondence: kmccall@bu.edu
https://doi.org/10.1016/j.xpro.2024.103017

SUMMARY

Vacuole formation is a key hallmark of age-dependent neurodegeneration in the *Drosophila* brain. Here, we present a protocol to analyze 3D neurodegenerative vacuoles in the whole-mount *Drosophila melanogaster* brain. We describe steps for whole-brain dissection, staining, 3D imaging, and z-stack image processing using Fiji ImageJ. We then detail procedures for annotating and 3D-reconstructing neurodegenerative vacuoles with WEBKNOSSOS and Python, and performing statistical analysis in Python. This protocol enables measurement of parameters such as the number and volume of each vacuole.

For complete details on the use and execution of this protocol, please refer to Elguero et al.[1]

BEFORE YOU BEGIN

We have established a comprehensive and reproducible image analysis protocol for studying neurodegenerative vacuoles in the whole-mount fly brain, utilizing open-source software – Fiji,[2] WEBKNOSSOS[3] and Python[4] (refer to the key resources table). WEBKNOSSOS, a web-based tool renowned for skeleton and volume annotation, has been widely employed in annotating connectomics datasets.[5,6] While WEBKNOSSOS is not entirely free, the 50 GB storage provided by its basic plan suffices for the purposes outlined in this paper. This section provides step-by-step guidance, including the installation of Fiji ImageJ and Python, WEBKNOSSOS account creation, and image conversion to RGB color (if required).

Adult whole-brain dissection, staining and mounting, and 3D microscopy imaging

Timing: 3 days

Prior to beginning, brain images are necessary. Detailed procedures for whole-mount fly brain staining and imaging are described in Behnke et al., 2021.[7] Whole brains of aged flies were dissected, stained with rhodamine phalloidin and DAPI to identify neurodegenerative vacuoles as regions devoid of both signals, and imaged from the anterior to posterior end using 2-photon or confocal microscopy. We have found that 2-photon microscopy offers advantages in efficiency and ease for capturing images of the entire fly brain. The higher penetration ability of 2-photon microscope eliminates the need for intensity compensation settings required by confocal microscopy. Image files generated by most confocal and 2-photon microscopy software systems can be directly opened in Fiji ImageJ, though some may require installation of a specific plugin corresponding to the imaging system. In our case, the 2-photon microscopy software bundled with an outdated imaging

acquisition system necessitated the installation of a specific plugin called "Prairie reader" in Fiji ImageJ (described in Elguero et al., 2023[1]). While this may not be a standard requirement in most labs, Fiji proves versatile in handling various image file types and converting them into RGB or 8-bit gray scale format for annotation in WEBKNOSSOS. Only use the special characters that can be recognized by python in the file names, such as underscore.

Software installation and account creation

Timing: <30 min

1. Fiji ImageJ Installation.
   a. Download and install Fiji ImageJ following the instructions from https://imagej.net/software/fiji/downloads.

   Note: Depending on the imaging system and software, installation of a specific plugin in Fiji ImageJ may be required.

2. WEBKNOSSOS Account Creation.
   a. Create a free account for WEBKNOSSOS from https://webknossos.org/. The default basic plan provides 50 GB of free storage.
3. Python installation.
   a. Download and install Python version 3.11.5 following instructions from https://www.python.org/downloads/.

Image conversion to RGB-color or 8-bit grayscale type using Fiji

Timing: < 2 min/file

Convert z-stacks acquired from confocal microscopes to RGB stacks or 8 bit grayscale stacks in the TIFF format for use with WEBKNOSSOS for annotation. We have provided manual and automatic methods for image conversion.

4. Manually convert raw images into the RGB-color or 8-bit grayscale format individually.
   a. Merge channels (phalloidin and DAPI channels).
      i. Open the file by dragging and dropping it in ImageJ.
      ii. Select "Image" > "Color" > "Merge channels".
      iii. Select the channel and place it in the designated row. For example, the phalloidin channel goes into C1 (red) and the DAPI channel goes into C5 (cyan). Check "Create composite" and "Keep source images" (Figure 1A).
      iv. Click "OK".

   Note: Adjust the brightness of each channel as needed by clicking "Image > Adjust > Brightness/Contrast" and "OK".

   b. Change file type of composite image to "RGB color". Select "Image > Type > RGB Color" (Figure 1B).
   c. Save image as Tiff. Select "File > Save As >Tiff" (Figure 1C).
5. Alternatively, automatically convert multiple raw images through batch processing using the FIJI macro provided in the GitHub repository (maketiff.ijm) (https://github.com/McCallLabBU/neurodegeneration_vacuole_quantification/blob/main/code/maketiff.ijm). Upon running the macro, both RGB-color or 8-bit grayscale format outputs will be produced for each raw image file.

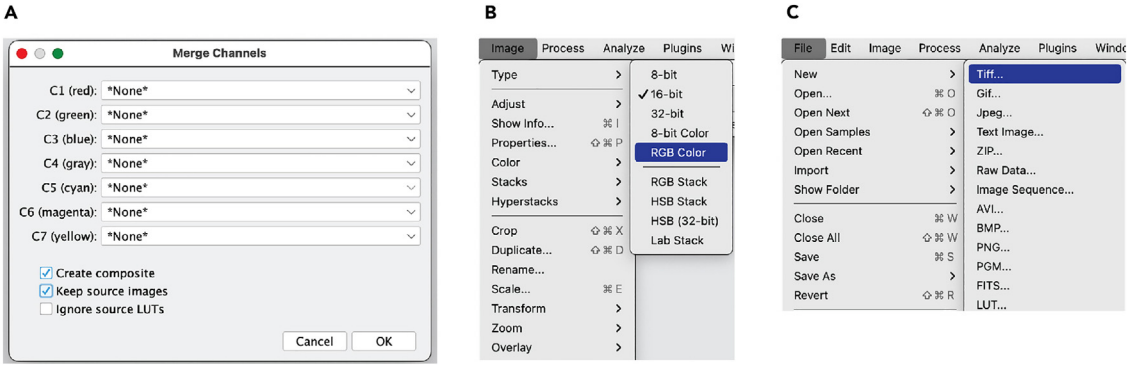


Figure 1. Conversion of raw data image into RGB-color or 8-bit grayscale format in Fiji ImageJ

(A) Merge phalloidin and DAPI channels.

(B) Change to RGB color or 8-bit.

(C) Save as TIFF.

Note: The macro can be modified to work with other common microscopy raw formats such as .oif.

a. Open the macro script maketiff.ijm in the FIJI script editor. Select "Plugins" > "macro" > "run".
b. Select the input folder containing all the raw images (e.g., .nd2 files).
c. Select the output destination folder to save the converted RGB-color or 8-bit grayscale images.
d. When the "Bio-Formats Import Options" window prompt appears, leave "Split channels" unchecked and click "OK".

⚠ CRITICAL: Only the necessary .nd2 files should be included in the input folder selected. Input .nd2 files must not contain any spaces or special characters other than underscores in the filename. An example of an ideal input filename is - best_input_file_123.tif.

KEY RESOURCES TABLE

REAGENT or RESOURCE	SOURCE	IDENTIFIER
Chemicals, peptides, and recombinant proteins		
Rhodamine phalloidin 1:100	Invitrogen	Cat#R415
VECTASHIELD PLUS antifade mounting medium with DAPI	Fisher Scientific	Cat#NC1848444
Deposited data		
Raw and analyzed data	This paper	
Experimental models: Organisms/strains		
Drosophila: UAS-LexA RNAi	Bloomington *Drosophila* Stock Center	RRID:BDSC_67947
Drosophila: UAS-drpr RNAi	Bloomington *Drosophila* Stock Center	RRID:BDSC_36732
Drosophila: *repo-GAL4/TM3*	Bloomington *Drosophila* Stock Center	RRID:BDSC_7415
Software and algorithms		
FIJI/ImageJ	Schindelin et al., 2012[2]	https://imagej.net/Fiji
WEBKNOSSOS	Boergens et al., 2017[3]	https://webknossos.org
Python version 3.11.5	Python Software Foundation	https://www.python.org

STEP-BY-STEP METHOD DETAILS

Here, we describe a step-by-step methodology for analyzing the number and volume of neurodegenerative vacuoles in the whole-mount fly brain. This process spans the conversion of raw data

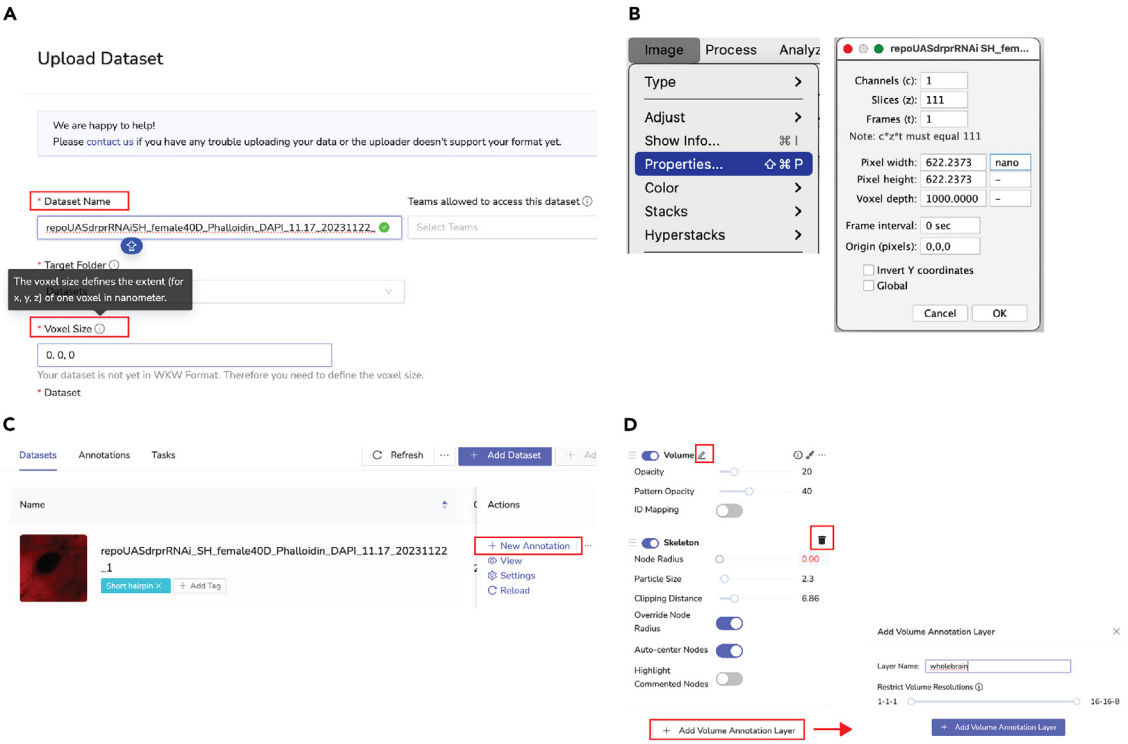


Figure 2. Dataset uploading and volume layer creation in WEBKNOSSOS

(A) Upload dataset and enter dataset name and voxel size.

(B) Find voxel size in Fiji.

(C) Add annotation for each brain.

(D) Create and rename 2 volume layers for vacuole and whole brain annotations.

and the segmentation of vacuoles, to the export, analysis, and visualization of vacuole feature vectors in Python. Areas devoid of staining are identified as vacuoles. First, using WEBKNOSSOS, vacuoles and whole brain is annotated, and their qualitative and quantitative information is exported as 3D meshes or csv files. Subsequently, statistical analysis is performed in Python, using a custom notebook described herein or researcher's preferred spreadsheet software.

Vacuole annotation using WEBKNOSSOS

Timing: 10–60 min/brain

This section of the protocol describes the annotations of individual vacuoles in WEBKNOSSOS. WEBKNOSSOS allows convenient annotations through "Quick Select" and "interpolate" tools, and annotation arrangement via grouping function.

1. Upload each Tiff file into WEBKNOSSOS.
   a. Select "Dashboard > Datasets".
   b. Drag the file into the upload window, and the Dataset Name field will be automatically populated with the file name by default (Figure 2A). Dataset Name can be modified according to the user's preference.
   c. Voxel size. This information can be obtained in Fiji (Figure 2B).
      i. Open the file in Fiji.
      ii. Select "Image > Properties". Change unit to "nanometers".
      iii. Enter the information into WEBKNOSSOS, e.g., 622.2373, 622.2373, 1000.

d. Click "Upload".
2. Select "Dataset" and find the uploaded file (Figure 2C).
3. Click "+ New Annotation" (Figure 2C).
4. Create 2 volume layers, one for vacuole annotation and another for whole brain annotation. On the left side of the annotation window, there are two default layers, Volume and Skeleton layer (Figure 2D).
   a. Rename the Volume layer as "Vacuoles".
   b. Click "+ Add Volume Annotation Layer" and rename it as "WholeBrain" (Figure 2D).
   c. Delete the Skeleton layer as it is unnecessary.
5. Vacuole annotation.
   a. Click on the Vacuoles layer on the left.
   b. Annotate the first vacuole. The first annotation will be automatically labeled as Segment 1. Scan through the entire brain, starting from the anterior end, and annotate each vacuole from the slice initially displayed using either the "Brush", "Trace" or "Quick Select" tool (Figure 3A).

⚠ CRITICAL: Start vacuole annotation prior to whole brain annotation to prevent the masking of vacuoles by the whole brain annotation. This sequence ensures that vacuole annotation can be successfully conducted afterward.

   c. Annotate an additional slice located approximately in the middle of a vacuole.
   d. Click "Interpolate" to annotate all slices between the current and the last annotated slice (Figure 3A). This facilitates a comprehensive annotation across the entire volume. Example vacuole annotation is shown in Figure 3B.

Note: Always verify the annotated slices using the "Interpolate" function, as it may not consistently annotate the entire region of interest (ROI) or may extend annotations beyond the intended ROI. Regular back-checking ensures accurate and precise annotation.

   e. Repeat step 5c & d until every slice of a vacuole is fully annotated.
   f. Annotate a new vacuole.
      i. Click "Create a new segment id" (Figure 3A).
      ii. Repeat step 5b, c, d and e.
   g. Repeat step 5f until all vacuoles are thoroughly annotated.

Optional: To enhance the organization of vacuoles within specific ROIs, such as the central brain versus optic lobes, and cortical rind versus neuropil, create segment groups for vacuoles in different ROIs. Right click "Roots" > "Create new group" > rename each group (Figure 3C). Annotated vacuoles/segments can be moved to the designated group later by right clicking on each group > "Move active segment here".

Whole brain annotation using WEBKNOSSOS

Timing: <10 min/brain

This step describes the annotation of the whole brain in WEBKNOSSOS.

6. Whole brain annotation.
   a. Click on the "WholeBrain" layer on the left.
   b. Annotate the first slice of the whole brain.
   c. Annotate another slice, approximately in the middle of the whole brain.

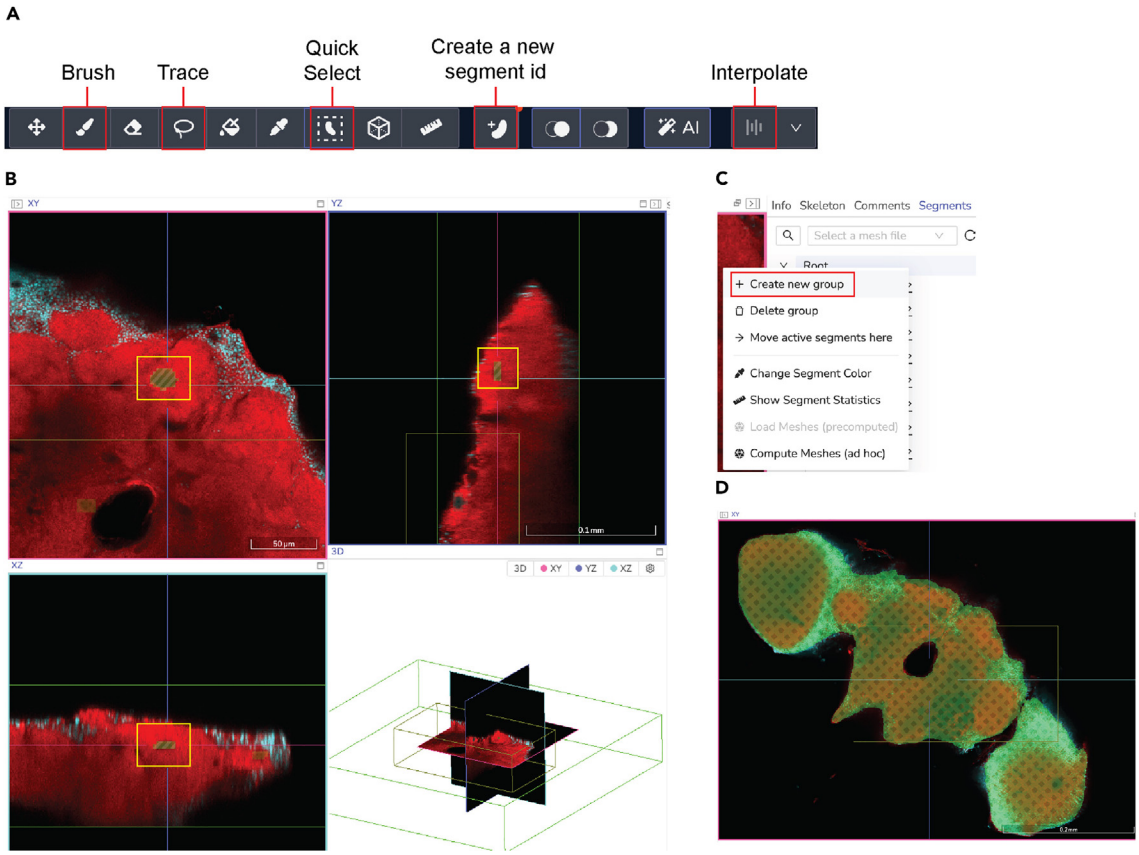


Figure 3. Vacuole annotation

(A) Introduction of WEBKNOSSOS tool bar.

(B) Example annotation of one vacuole.

(C) Organize annotations into different groups (optional).

(D) Example annotation of one whole-brain slice.

d. Click "Interpolate" to ensure all slices between the current and the last annotated slice are included. Example whole-brain slice annotation is shown in Figure 3D.

Note: Always cross-check the annotated slices using the "Interpolate" function. Ensure that physiological structures, such as esophagus and air sacs, are not mistakenly annotated.

Extract vacuole features for downstream analysis and visualization

Timing: <5 min/brain

This section describes how to convert all the annotations into 3-dimensional meshes and export them as either .stl files or .csv files into an organized directory structure for downstream analysis and visualization in Python.

7. Compute meshes for vacuoles.
   a. Open "Vacuoles" layer
   b. Right click "Root" > "Compute Meshes (ad hoc)" (Figure 4A). Example vacuole meshes are shown in Figure 4B.
   c. Download meshes once the computation is complete. Right click "Root" > "Download Meshes" (Figure 4C).

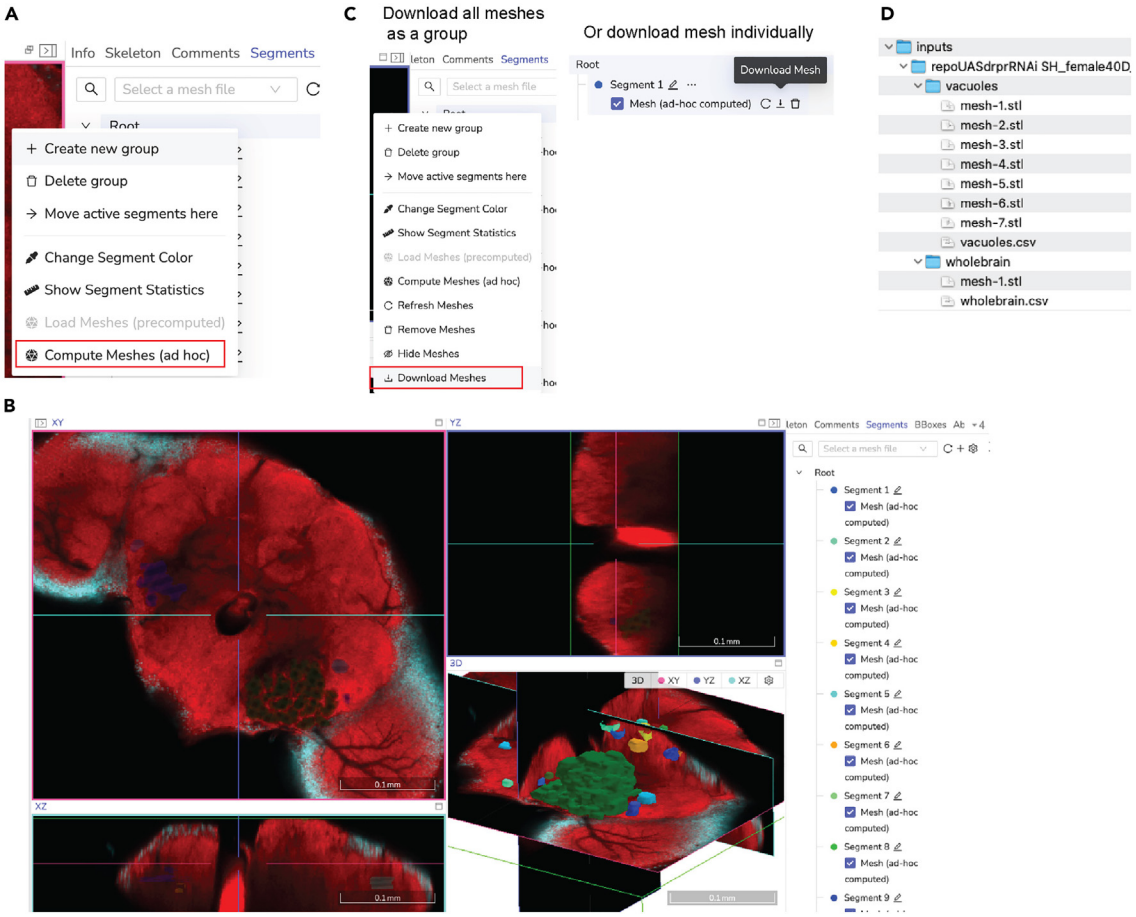


Figure 4. 3D meshes computation

(A) Compute all meshes.

(B) Example of complete meshes of vacuoles.

(C) Download all meshes at once or download each mesh individually.

(D) Folder organization for the convenience of analysis in Python. Save downloaded meshes in this format: "inputs" > "your desired experimental group" > "vacuoles" and "wholebrain".

Alternatives: If all annotations can't be downloaded as a group, download individual annotations one by one: click the download icon next to the computed mesh (Figure 4C).

d. Create folders for downloaded meshes for each brain. Inputs > brain name > vacuoles > each vacuole (keep consistent naming format) (Figure 4D).

8. Alternatively, export vacuole features as .csv files.
   a. Right click "Root" > "Show segment statistics" > "Export to CSV" (Figure 5).

Exploratory data analysis, hypothesis testing, and visualization in python

Timing: 15 mins

This section describes statistical analysis and visualization of vacuole data in Python. Once the .csv files are generated, further statistical analysis can be conducted using any preferred spreadsheet software. Alternatively, we have provided Python notebooks for the analysis as an option. The workflow to extract and visualize the distribution of the number of vacuoles and the percent of whole brain volume occupied by vacuoles, and the distribution of raw and normalized vacuole volumes

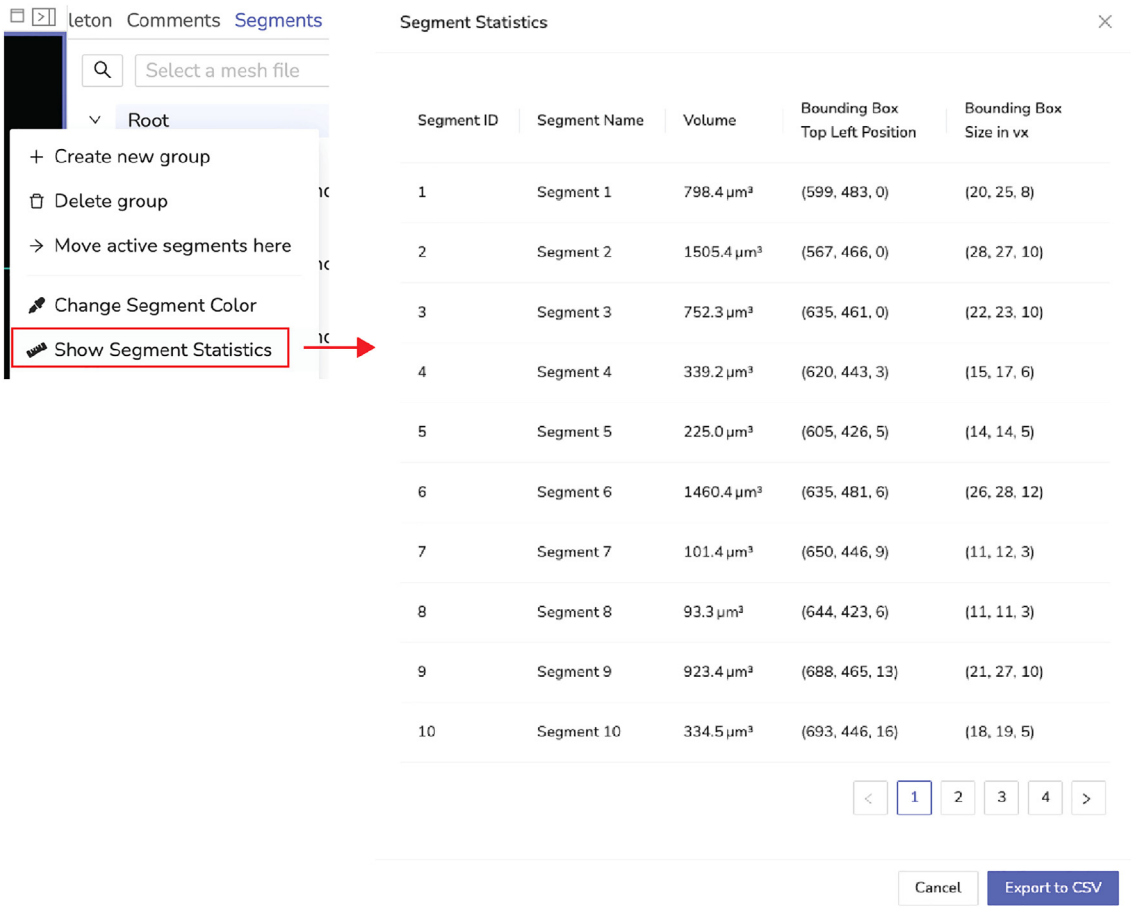


Figure 5. Segment statistics in WEBKNOSSOS

Number, volume and position information of individual annotations can be exported to .csv file from WEBKNOSSOS.

(nm^3) in each brain sample across experimental groups is described below. Additionally, one-sided independent samples t-test between experimental groups for the aforementioned distributions is also implemented. Methods for both .stl and .csv files are described.

9. Input file organization.
   a. Before beginning downstream analysis, organize the exported .stl or .mesh files as shown in Figure 4D.

   Note: Each sample must have its own directory, within which .stl or .csv files from vacuoles and whole brain are organized separately.

   Note: Sample name ("condition_name" in the directory hierarchy specified above) must begin with the experimental group name, followed by a unique sample or replicate identifier.

   b. Ensure that the directory names are exactly as specified above for "inputs", "vacuoles" and "wholebrain" and that there are no spaces in the file or directory names.

   Note: Underscores must be used to separate group or replicate identifiers.

10. Download the python analysis notebook and set up the analysis environment.
    a. In a terminal window, navigate to a directory of choice and clone the GitHub repository.

```
>cd /path/to/your/directory/

>git clone https://github.com/McCallLabBU/neurodegeneration_vacuole_quantification.git

>cd neurodegeneration_vacuole_quantification
```

Alternatives: Download the repository as a zip file from the link provided (See data and code availability section) and unzip it in a directory of choice.

b. Create a Python3 virtual environment.

```
>python3 -m venv vacuolequant

>source vacuolequant/bin/activate
```

c. Install necessary packages.

```
>pip3 install ipykernel

>pip3 install jupyter

>pip3 install numpy-stl

>pip3 install matplotlib

>pip3 install seaborn

>pip3 install numpy

>pip3 install pandas

>pip3 install scipy

>pip3 install scipy
```

Alternatives: Use the requirements.txt file provided to install all necessary packages at once:

```
>pip3 install -r requirements.txt
```

d. Create a Jupyter notebook kernel.

```
>python3 -m ipykernel install --name=vacuolequant --user

>jupyter notebook
```

11. Open the appropriate analysis notebook.

Note: If using .stl files for analysis, make a copy of the notebook quantify_vacuoles_fromMesh_LiuBandyadka2024.ipynb. If using .csv files for analysis, duplicate the quantify_vacuoles_fromWebknossosexport_LiuBandyadka2024.ipynb notebook. Open the duplicated notebook.

12. Replace paths to input and output directories and experimental info in the "Set up directories and experiment info" section.

Note: Lines that end with the comment "#replace" must be edited appropriately in all code cells, making sure to follow correct python syntax.

13. Run all the cells in the notebook.
    a. Visualizations of the distribution of number of vacuoles, volume (raw and normalized) of vacuoles, and percent of whole brain volume occupied by vacuoles will appear below the appropriate code sections and will also be saved as PDFs in the output path specified.
    b. One-sided independent samples t-test is run for the distributions for the experimental groups after the visualization.
    c. The test statistic and p-value are also output below the appropriate code cell.
    d. The consolidated vacuole metrics from all samples is exported as a .csv file to the output path specified.

Note: We first implemented a method in Python to extract vacuole features from .stl mesh files, which was used in Elguero et al., 2023.[1] The WEBKNOSSOS team has since introduced the new segment statistics feature, which allows the export of annotated features as .csv files. Here we provide instructions for extracting and visualizing vacuole features using both .stl and .csv files. Notably, there are minor differences in the absolute volume measurements between the two methods. However, the relative differences between experimental groups are comparable (Figure 6G).

EXPECTED OUTCOMES

Flies with the phagocytic receptor, Draper (Drpr), knocked down in glia have been previously reported to display more neurodegenerative vacuoles compared to controls.[8] Our analysis shows that brains lacking *drpr* exhibit larger areas of vacuolization (Figure 6). Loss of glial *drpr* led to higher number of neurodegenerative vacuoles (Figures 6A and 6D) and the sum of vacuole volumes as a fraction of whole brain volume is significantly larger than that of the controls (Figures 6B and 6E). However, the volume of individual vacuoles in *drpr RNAi* flies is not significantly different than that of controls (Figures 6C and 6F). Brains lacking *drpr* contain areas of contiguous clusters of many vacuoles which are difficult to distinguish separately, thus we annotated them as one vacuole. Therefore, in Figures 6C and 6F, there are several data points of larger volumes. In addition, due to this annotation limitation, rather than the number of vacuoles, total vacuole volume as a fraction of whole brain volume (Figures 6B and 6E) is a more reliable proxy for neurodegeneration.

We also confirmed that vacuole volume statistics provided by WEBKNOSSOS showed comparable results with that of our Python pipeline (Figure 6G). Thus, for users without experience in coding, .csv files can be exported from WEBKNOSSOS and further statistical analysis can be performed using preferred software.

LIMITATIONS

We provide a comprehensive and reproducible pipeline using WEBKNOSSOS and Python to annotate and analyze vacuole number and volume in whole-mount fly brains displaying neurodegeneration. For samples with massive areas of vacuoles which are hard to separate individually, volume will be a better representation of neurodegeneration than number. Apart from number and volume, position of vacuoles in the brain is also an important aspect of neurodegeneration as damaged regions are highly correlated with dysfunction. However, our pipeline does not provide analysis of the positional information of vacuoles. If interested, people can develop further analysis of positions using .csv files exported from WEBKNOSSOS.

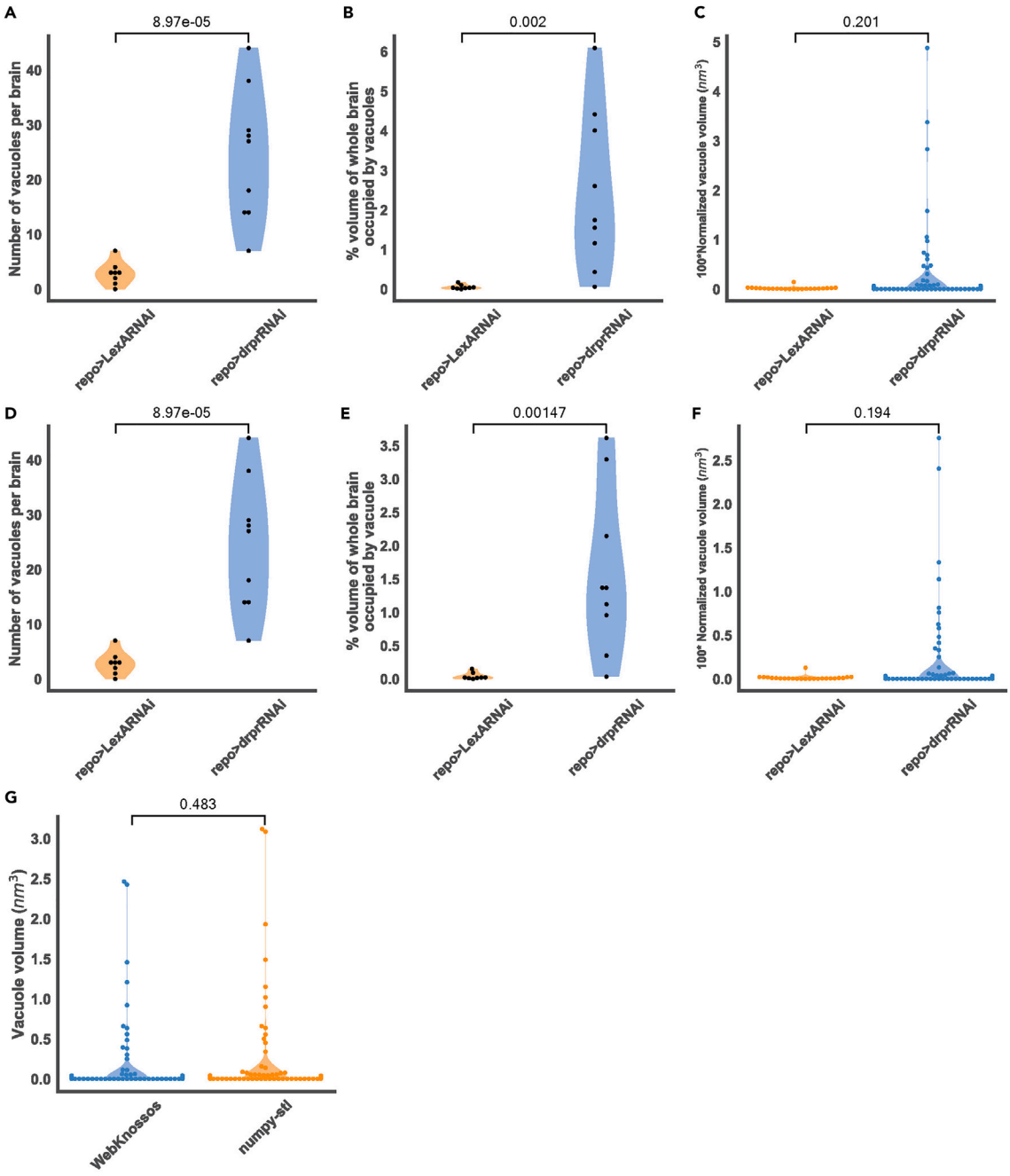


Figure 6. Statistical analysis of vacuole number and volume in *repo>LexARNAi* (*n* = 8) and *repo>drprRNAi* (*n* = 9) fly brains

(A–C) Data extracted from numpy-stl pipelines using Python. (A) Distribution of number of vacuoles identified per brain (p = 8.97e-05). (B) Distribution of percentage of whole brain volume occupied by vacuoles per brain (p = 0.002). (C) Distribution of individual vacuole volumes as a fraction of whole brain volume (p = 0.201).

(D–F) Data extracted from WEBKNOSSOS .csv file.

(D) Distribution of number of vacuoles identified per brain (p = 8.97e-05).

(E) Distribution of percentage of whole brain volume occupied by vacuoles per brain (p = 0.00147).

(F) Distribution of individual vacuole volumes as a fraction of whole brain volume (p = 0.194).

(G) Comparison of normalized vacuole volumes extracted from WEBKNOSSOS and numpy-stl (p = 0.483). One-sided independent samples t-test was performed using scipy Python package.

Statistical hypothesis testing for vacuole numbers and volume has been implemented for two experimental groups only but can be expanded in the future to include more than two groups and other tests using the scipy or statsmodels python packages.

TROUBLESHOOTING

Problem 1

Issues in installing Python in the software installation and account creation step 3.

Potential solution

Refer to several comprehensive, freely available guides provided on the Python website https://wiki.python.org/moin/BeginnersGuide/Download.

Problem 2

The brightness of images is too low after conversion in the Image conversion step 4&5.

Potential solution

Adjust the brightness of each channel of each raw image as needed in Fiji ImageJ before conversion. Remember to click "apply" after adjusting the brightness and save the images. Run the macro to convert raw images to RGB-color or 8-bit grayscale type.

Problem 3

Issues in running macro to convert raw images to RGB-color or 8-bit grayscale type in the image conversion step 5.

Potential solution

The filenames should not include special characters other than underscores. Remember to uncheck "Spilt channels" when the "Bio-formats Import Options" pops up.

Problem 4

The files are too big, or there is insufficient storage space in the WEBKNOSSOS in the vacuole annotation using WEBKNOSSOS step 1.

Potential solution

Convert the raw image to 8-bit grayscale to minimize the file size. After annotating and backing up annotations securely, the files can be deleted or archived.

Problem 5

There is significant variability in the brain volumes within each experimental group.

Potential solution

Make sure to enter the correct voxel size for each brain in the vacuole annotation using WEBKNOSSOS step 1c.

Problem 6

There are so many vacuoles clustering together that individual vacuoles can't be recognized and annotated separately in the vacuole annotation using WEBKNOSSOS step 5.

Potential solution

Annotate that area as one vacuole and use percentage of whole brain volume occupied by all vacuoles as a representation of neurodegeneration.

Problem 7

Can't download all meshes at once in the extract vacuole features for downstream analysis and visualization step 7.

Potential solution

Download each mesh individually.

RESOURCE AVAILABILITY

Lead contact

Further information and requests for resources and reagents should be directed to and will be fulfilled by the lead contact, Kim McCall, kmccall@bu.edu.

Technical contact

Questions about the technical specifics of performing fly brain staining and vacuole annotation should be directed to and will be answered by the technical contact, Guangmei Liu (gmliu@bu.edu). Questions about the technical specifics of the code should be directed to and will be answered by another technical contact, Shruthi Bandyadka (sbandya@bu.edu).

Materials availability

This study did not generate new unique reagents.

Data and code availability

All the python jupyter notebooks along with documentation have been deposited in the GitHub: https://github.com/McCallLabBU/neurodegeneration_vacuole_quantification; Zenodo: DOI: https://doi.org/10.5281/zenodo.10810496. Example annotated data (.stl and .csv files) are available as a compressed .zip file in the inputs folder of the GitHub repository. Example raw images (.tiff) are in the supplemental.

SUPPLEMENTAL INFORMATION

Supplemental information can be found online at https://doi.org/10.1016/j.xpro.2024.103017.

ACKNOWLEDGMENTS

We thank the Bloomington *Drosophila* Stock Center for providing fly strains. We thank NIGMS/NIH for funding (1 R35 GM127338).

AUTHOR CONTRIBUTIONS

Conceptualization, G.L. and S.B.; methodology, S.B.; investigation, G.L. and S.B.; writing, G.L. and S.B.; funding acquisition, K.M.; supervision, K.M.

DECLARATION OF INTERESTS

The authors declare no competing interests.

REFERENCES

1. Elguero, J.E., Liu, G., Tiemeyer, K., Bandyadka, S., Gandevia, H., Duro, L., Yan, Z., and McCall, K. (2023). Defective phagocytosis leads to neurodegeneration through systemic increased innate immune signaling. iScience *26*, 108052. https://doi.org/10.1016/j.isci.2023.108052.

2. 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.

3. Boergens, K.M., Berning, M., Bocklisch, T., Bräunlein, D., Drawitsch, F., Frohnhofen, J., Herold, T., Otto, P., Rzepka, N., Werkmeister, T., et al. (2017). WebKnossos: Efficient online 3D data annotation for connectomics. Nat. Methods *14*, 691–694. https://doi.org/10.1038/nmeth.4331.

4. Van Rossum, G., Drake, F.L., Harris, C.R., Millman, K.J., van der Walt, S.J., Gommers, R., Virtanen, P., Cournapeau, D., Wieser, E., Taylor, J., et al. (2009). Python 3 Reference Manual (Create Space).

5. Loomba, S., Straehle, J., Gangadharan, V., Heike, N., Khalifa, A., Motta, A., Ju, N., Sievers, M., Gempt, J., Meyer, H.S., and Helmstaedter, M. (2022). Connectomic comparison of mouse and human cortex. Science *377*, eabo0924. https://doi.org/10.1126/science.abo0924.

6. Song, K., Feng, Z., and Helmstaedter, M. (2023). High-contrast en bloc staining of mouse whole-brain and human brain samples for EM-based connectomics. Nat. Methods *20*, 836–840. https://doi.org/10.1038/s41592-023-01866-3.

7. Behnke, J.A., Ye, C., Moberg, K.H., and Zheng, J.Q. (2021). A protocol to detect neurodegeneration in *Drosophila melanogaster* whole-brain mounts using advanced microscopy. STAR Protoc. *2*, 100689. https://doi.org/10.1016/j.xpro.2021.100689.

8. Etchegaray, J.I., Elguero, E.J., Tran, J.A., Sinatra, V., Feany, M.B., and McCall, K. (2016). Defective phagocytic corpse processing results in neurodegeneration and can be rescued by TORC1 activation. J. Neurosci. *36*, 3170–3183. https://doi.org/10.1523/JNEUROSCI.1912-15.2016.