ACVA: Automated Clear Volume Analysis

Rong Chen

August 8, 2025

1 Introduction

This software package, Automated Clear Volume Analysis (ACVA), aims to automatically or semi-automatically analyze clear volume imaging data of rodent brains.

2 Installation

ACVA is based on Python 3. It is strongly recommended that users create a virtual environment, as this is essential for maintaining a clean and isolated project setup. The installation process is as follows:

- Install Anaconda.
- Create a virtual environment. For Windows users, they can use Anaconda Powershell. For Linux users, the shell can be bash in a XTerm. Insider a shell such as bash, type this: conda create -n env_acva python=3.10
- Activate the virtual environment: conda activate env_acva
- Install required packages: conda install numpy pandas matplotlib scikit-learn scipy conda-forge::scikit-image conda-forge::SimpleITK imageio
- Download the ACVA package and unzip it to a folder. This package also has a test dataset.
- Test the installation. Open a shell (Windows users can use the Anaconda PowerShell), activate the virtual environment, navigate to the folder containing the package, and then run the following command: *python test_seg_particle.py* . The output should be 'Happy ending'.

3 Data

The imaging data are organized with the the Brain Imaging Data Structure (BIDS) standard. All grayscale images for a single channel are saved in a folder.

A sample dataset is in the data_test folder of the ACVA package. This dataset has one channel: neuron_nuclei.

Consistent file naming: Use sequence numbering such as Z0001, Z0002, ... (not Z1, Z2, ...). This convention is important because the software is Python-based and reads all image files in a folder. If filenames like Z1, Z2, etc. are used, the slice order in the software could be random.

Don't merge tiff files as a single stack. This is because this practice will create a huge file that is difficult to load into memory for computation.

For both folder and file names, avoid using white spaces. Many shells do not handle them well, which can lead to unpredictable behavior.

4 Usage

4.1 Define your project

Before running the analysis, you need to define a project by creating a csv file such as exp.csv. This file has two columns. The first column is data source and the second is related operation. It has a header line.

- Line 1 specifies the project's home folder with operation set to home (or any other string). **For home folder, we must use the absolute path instead of a relative path**.
- Line 2 defines the channel for particle detection. The first field in this line is the channel name, and the second field should be "seg_particle".

An example experiment file is exp_sample.csv.

4.2 Hyperparameter tuning

When a user analyzes the data for the first time, the optimal hyperparameters are unknown and must be determined through a search.

Open a shell, activate the virtual environment, and run the following command: python acva_seg_particle_main.py exp.csv. This command is referred as the main command.

Notice that exp.csv is project specific. For example, if the experiment file is exp_cfos.csv, then the main command is python acva_seg_particle_main.py exp_cfos.csv.

This will search for the optimal hyperparameters and save them to the file conf_seg_particle.csv. This file is referred to as the configuration file. ACVA will create a folder for the channel. For example, if channel is cFos, it will create a new folder cFos_res under the project home directory. The configuration file is saved inside the channel result folder.

The hyperparameter tuning process uses a subset of the dataset due to computational cost. For an image stack, it uses the middle block.

4.3 Full run after hyperparameter tuning

Once the configuration file is created, you can run the full analysis by running the main command again. The particle analysis results are in file all_objects.csv inside the channel result folder.

If the configuration file is not available, the user creates it by running the main command once, and then performs the full analysis by running the main command again. Therefore, the user runs the main command twice.

If the configuration file is available, the main command will always use it for full analysis, and the automated parameter tuning process will be disabled. If a user wants to manually tune parameters, they can modify the configuration file, run the full analysis, and determine the optimal parameters by inspecting the results.

5 Visualization

5.1 3D rendering of particles

For particle analysis, ACVA generates a file all_objects.csv. We can use ParaView to generate 3D rendering of the point cloud.

- Open ParaView, Click File → Open, select the csv file.
- When the "CSV Reader" dialog appears, check that "Have Headers" is ticked and make sure it reads headers correctly, then click "Apply".
- Convert it to a point cloud. Go to Filters → Alphabetical → Table To Points. In the TableToPoints filter, set x, y, z. Then click "Apply". Now the point cloud appears.
- For improved effect, right click the point cloud; then choose Representation → "Point Gaussian" and "Color By" → Volume.
- Save the whole project as File → "Save State". By default, it does not embed your data and just references the files by their path.