Calcium Imaging Analysis Scripts for MATLAB – User Guide

Tampere University, Biophysics of the Eye Group

Juhana Sorvari
juhana.sorvari@tuni.fi

Last updated 31.12.2020

Preface

This document is serves as a basic guide to using the scripts released as supporting information with the publication *Analysis of ATP-induced Ca^{2+} Responses at Single Cell Level in Retinal Pigment Epithelium Monolayers*, Adv. Exp. Med. Biol. 2019;1185:525-530. The scripts themselves are not limited to calcium imaging or intensity analysis. They can be used to analyse any desired phenomena, as long as the data meets the requirements for the scripts (see chapter 2 of this guide).

This guide, as well as the latest version of the scripts, can be downloaded from GitHub at https://github.com/teemui/Ca-analysis.

7.10.2019
Tampere, Finland
Juhana Sorvari
juhana.sorvari@tuni.fi

Table of Contents

1. Installing the scripts

Included files

The script package consists of the following files:

intensityAnalysis_start.m

- the master file that starts the main menu of the program

intensityResponse.m

- a class definition file for the *intensityResponse* class that contains its parameters and methods

editDatabase.m

- a function that is called by several scripts to fetch datasets from a database or to merge separate databases into one database

readData.m

- a script that reads the intensity data from a spreadsheet to Matlab and saves it to a database (.mat file)

analysisLoop.m

- a script that runs the actual intensity analysis for the dataset that is selected

curveBrowser.m

- a script for plotting and viewing the analysed intensity curves or comparing responses of the same cell in different measurements

clusterAnalysis.m

- a script for running the clustering algorithms to group similar intensity responses

plotLocationData.m

- a script that visualizes the grouping done by clusterAnalysis.m over the original ca-image using the same colors as clusterAnalysis.m

statisticsBlock.m

- a script for making data visualizations of the analysed response parameters (boxplots or scatter plots at the moment)

inputsdlg.m

- a modified input dialog script from Matlab file exchange, included by the permission in the licence file and with the appropriate copyright notice

Installation

Copy all the .m files to your Matlab root folder (usually \Documents\MATLAB\ in Windows). If the files are located elsewhere, you have to add the folder to your Matlab path by locating it in the Matlab "Current Folder" window and right clicking it. Then select "Add to path" -> "Selected folders and subfolders". **Please note:** Always back up the scripts before installing new versions, because backwards compatibility cannot be guaranteed! It is also a good idea to finish analysing one dataset/database before switching to a new version.

Also note that following toolboxes need to be installed in addition to Matlab base installation:

curve fitting toolbox
signal processing toolbox
statistics toolbox

2. Preparing the data for analysis

Preparations in ImageJ

First, crop the image stack to a region of interest (ROI) before outlining single cells. If there is a lot of drift, run a stabilization algorithm (e.g. stackreg or similar). To get good contrast, take a Z-project of the whole stack (Image -> Stacks -> Z-Project, use any projection type that gives the best result, usually standard deviation works fine). Be sure to adjust contrast (Image -> Adjust -> Brightness/Contrast) to a suitable level. On this image, outline the cells using for example the circle tool (Fig 2.1). First, open the ROI manager (Analyze -> Tools -> ROI manager) and then start outlining cells from the image and press T to add them to the ROI list. Also tick the "Show all" box to see all the cells in the image.

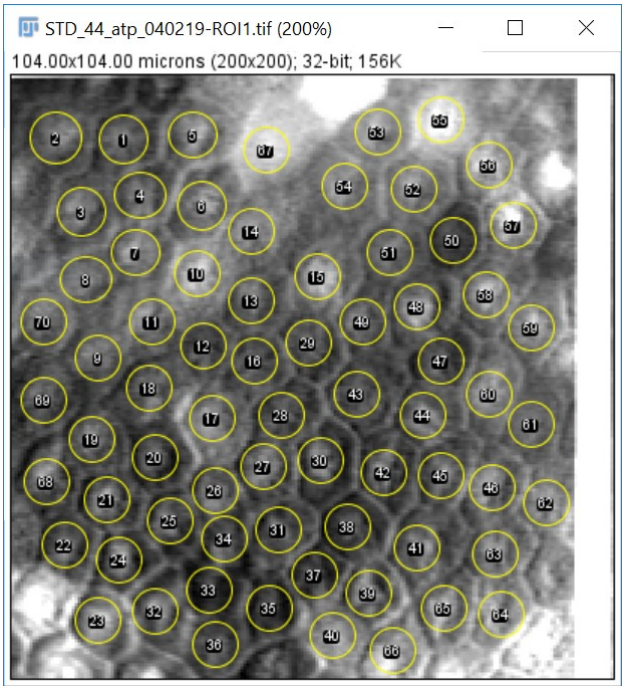


Figure 2.1 Outlined cells on the Z-projected image

Outline enough cells to get a good population (e.g. 70-100 cells per ROI) and then save the ROI list in the ROI manager (More -> Save). Now, select the image stack that you used to make the Z-project and tick “Show All” again in the ROI manager. You should now see all the outlined cells on the stack.

Next, select *More -> Multi Measure* in the ROI manager and check that *Measure all slices* and *One row per slice* are ticked. When you get the results sheet for the first time, open *Results -> Set Measurements* and select *Area, Center of mass, Mean gray value* and *Stack position*. Close the results table and open it again with multi measure to make sure it updates. After this, you should see the following table with four columns per cell: *Area*, *Mean*, *XM* and *YM* (Fig 2.2). The first column on the right is the data point (frame) number.

Results

File Edit Font Results

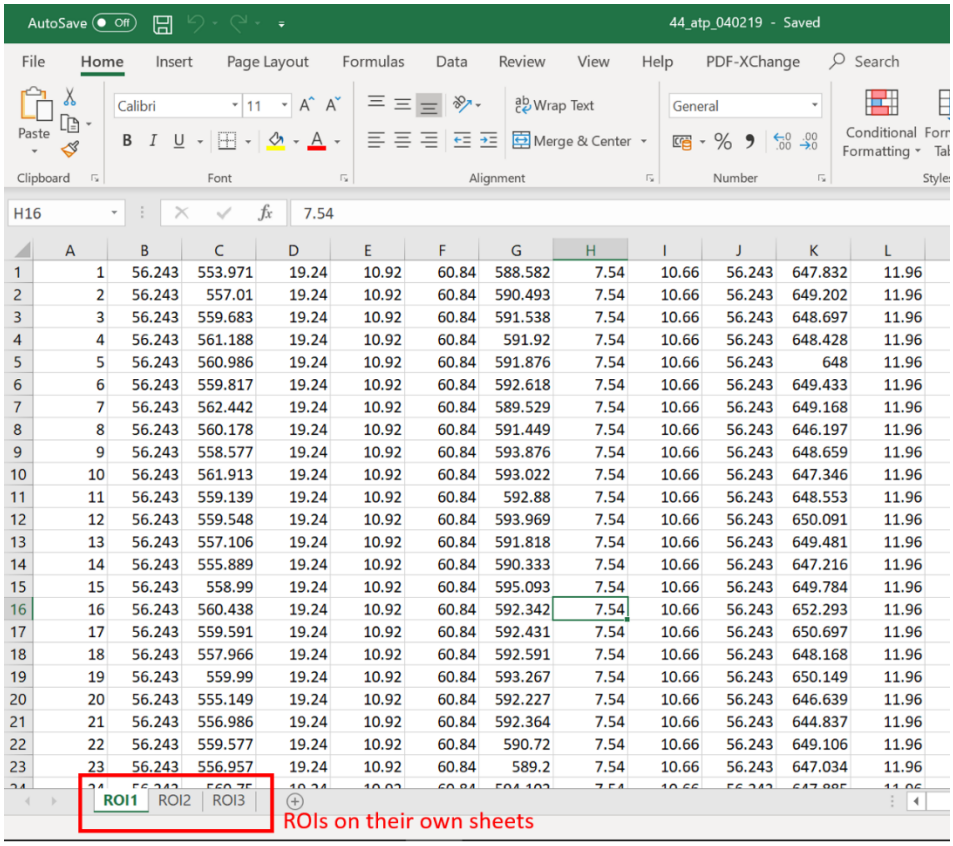
	Area1	Mean1	XM1	YM1	Area2	Mean2	XM2	YM2	Area3	Mean3	XM3	YM3	Area
1	18.684	1159.721	7.214	7.369	25.115	1190.232	13.062	4.510	25.115	891.224	12.685	10.769	20.3
2	18.684	1153.239	7.196	7.370	25.115	1199.086	13.054	4.526	25.115	895.606	12.705	10.767	20.3
3	18.684	1174.598	7.214	7.393	25.115	1210.824	13.057	4.539	25.115	889.271	12.710	10.770	20.3
4	18.684	1166.261	7.209	7.388	25.115	1201.942	13.063	4.504	25.115	894.299	12.707	10.746	20.3
5	18.684	1176.181	7.212	7.369	25.115	1204.860	13.057	4.500	25.115	898.828	12.720	10.758	20.3
6	18.684	1172.553	7.216	7.375	25.115	1205.721	13.051	4.531	25.115	896.946	12.717	10.764	20.3
7	18.684	1174.126	7.211	7.384	25.115	1199.860	13.054	4.501	25.115	900.867	12.691	10.768	20.3
8	18.684	1166.030	7.204	7.370	25.115	1192.559	13.067	4.508	25.115	899.064	12.707	10.759	20.3
9	18.684	1173.764	7.211	7.370	25.115	1197.194	13.056	4.506	25.115	894.763	12.715	10.752	20.3
10	18.684	1187.520	7.219	7.372	25.115	1188.135	13.059	4.526	25.115	897.807	12.713	10.769	20.3
11	18.684	1172.141	7.217	7.375	25.115	1194.064	13.063	4.513	25.115	894.940	12.726	10.766	20.3
12	18.684	1176.033	7.206	7.372	25.115	1201.049	13.060	4.533	25.115	891.649	12.702	10.765	20.3
13	18.684	1171.372	7.221	7.361	25.115	1183.208	13.092	4.529	25.115	896.864	12.708	10.765	20.3
14	18.684	1163.900	7.216	7.364	25.115	1208.196	13.062	4.513	25.115	885.221	12.692	10.757	20.3
15	18.684	1173.427	7.215	7.368	25.115	1210.690	13.046	4.509	25.115	889.535	12.702	10.761	20.3
16	18.684	1158.349	7.219	7.383	25.115	1208.678	13.055	4.516	25.115	899.417	12.717	10.755	20.3
17	18.684	1165.756	7.201	7.376	25.115	1200.464	13.048	4.527	25.115	897.078	12.705	10.782	20.3
18	18.684	1152.322	7.226	7.381	25.115	1206.806	13.056	4.508	25.115	903.703	12.713	10.768	20.3
19	18.684	1146.611	7.217	7.366	25.115	1199.198	13.065	4.525	25.115	895.495	12.711	10.764	20.3
20	18.684	1170.357	7.207	7.393	25.115	1203.800	13.058	4.515	25.115	901.333	12.716	10.748	20.3
21	18.684	1159.279	7.209	7.369	25.115	1191.905	13.065	4.523	25.115	899.873	12.719	10.764	20.3
22	18.684	1178.553	7.229	7.379	25.115	1199.957	13.067	4.512	25.115	893.622	12.693	10.758	20.3
23	18.684	1166.538	7.208	7.358	25.115	1215.914	13.057	4.519	25.115	917.645	12.718	10.770	20.3
24	18.684	1149.432	7.202	7.378	25.115	1201.972	13.061	4.508	25.115	895.703	12.702	10.763	20.3
25	18.684	1165.671	7.222	7.382	25.115	1213.492	13.075	4.529	25.115	902.991	12.705	10.759	20.3
26	18.684	1150.319	7.212	7.375	25.115	1215.804	13.061	4.527	25.115	896.903	12.696	10.771	20.3
27	18.684	1151.548	7.230	7.378	25.115	1210.787	13.048	4.515	25.115	895.265	12.716	10.769	20.3
28	18.684	1166.648	7.196	7.368	25.115	1208.778	13.066	4.516	25.115	906.273	12.715	10.767	20.3
29	18.684	1166.138	7.207	7.381	25.115	1206.714	13.053	4.519	25.115	909.652	12.718	10.751	20.3

Figure 2.2 The results sheet from a single ROI

Copy the results sheet to an excel spreadsheet and repeat the process for the next ROIs.

Preparing the spreadsheet for MATLAB

Last preparation step is to make sure the spreadsheet is in the correct form for the scripts. One xlsx-file represents an original image stack, which is divided into ROIs (cropped image stacks). Each ROI results sheet from ImageJ should be copied to its own sheet in the xlsx-file (Fig 2.3).



	A	B	C	D	E	F	G	H	I	J	K	L
1	1	56.243	553.971	19.24	10.92	60.84	588.582	7.54	10.66	56.243	647.832	11.96
2	2	56.243	557.01	19.24	10.92	60.84	590.493	7.54	10.66	56.243	649.202	11.96
3	3	56.243	559.683	19.24	10.92	60.84	591.538	7.54	10.66	56.243	648.697	11.96
4	4	56.243	561.188	19.24	10.92	60.84	591.92	7.54	10.66	56.243	648.428	11.96
5	5	56.243	560.986	19.24	10.92	60.84	591.876	7.54	10.66	56.243	648	11.96
6	6	56.243	559.817	19.24	10.92	60.84	592.618	7.54	10.66	56.243	649.433	11.96
7	7	56.243	562.442	19.24	10.92	60.84	589.529	7.54	10.66	56.243	649.168	11.96
8	8	56.243	560.178	19.24	10.92	60.84	591.449	7.54	10.66	56.243	646.197	11.96
9	9	56.243	558.577	19.24	10.92	60.84	593.876	7.54	10.66	56.243	648.659	11.96
10	10	56.243	561.913	19.24	10.92	60.84	593.022	7.54	10.66	56.243	647.346	11.96
11	11	56.243	559.139	19.24	10.92	60.84	592.88	7.54	10.66	56.243	648.553	11.96
12	12	56.243	559.548	19.24	10.92	60.84	593.969	7.54	10.66	56.243	650.091	11.96
13	13	56.243	557.106	19.24	10.92	60.84	591.818	7.54	10.66	56.243	649.481	11.96
14	14	56.243	555.889	19.24	10.92	60.84	590.333	7.54	10.66	56.243	647.216	11.96
15	15	56.243	558.99	19.24	10.92	60.84	595.093	7.54	10.66	56.243	649.784	11.96
16	16	56.243	560.438	19.24	10.92	60.84	592.342	7.54	10.66	56.243	652.293	11.96
17	17	56.243	559.591	19.24	10.92	60.84	592.431	7.54	10.66	56.243	650.697	11.96
18	18	56.243	557.966	19.24	10.92	60.84	592.591	7.54	10.66	56.243	648.168	11.96
19	19	56.243	559.99	19.24	10.92	60.84	593.267	7.54	10.66	56.243	650.149	11.96
20	20	56.243	555.149	19.24	10.92	60.84	592.227	7.54	10.66	56.243	646.639	11.96
21	21	56.243	556.986	19.24	10.92	60.84	592.364	7.54	10.66	56.243	644.837	11.96
22	22	56.243	559.577	19.24	10.92	60.84	590.72	7.54	10.66	56.243	649.106	11.96
23	23	56.243	556.957	19.24	10.92	60.84	589.2	7.54	10.66	56.243	647.034	11.96

Figure 2.3 The spreadsheet file with correct naming convention. Each ROI should be on its own sheet.

The naming of the sheets (and the xlsx-file itself) should be as follows in order for Matlab to be able to use them as variables:

- all names should begin with a letter
- no spaces or special characters (use underscores to replace spaces and hyphens)

Note: Check that the numeric data is in correct decimal form so that Excel interprets them as numbers. Use find/replace to change . to , if needed.

3. Reading the data to MATLAB

Reading the data to new or existing database

Run the master script by typing *intensityAnalysis_start* to the command line. This boots up the main menu (Fig 3.1)

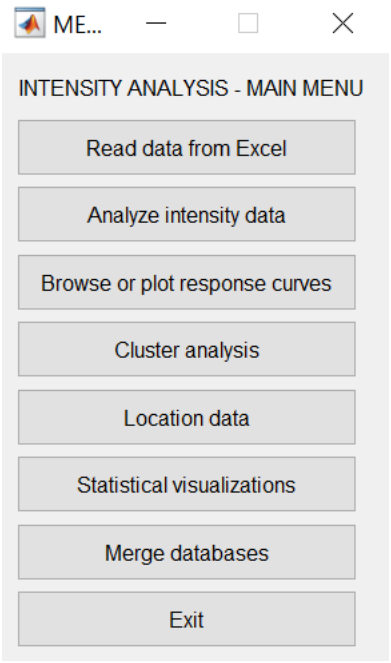


Figure 3.1 The main menu of the intensity analysis scripts

Start by selecting *Read data from Excel*. A window pops up allowing you to browse for the spreadsheets you want to import. You can select multiple files at a time. Next, the program asks you to name each dataset (= one spreadsheet file) as you want it to appear in the database (Fig 3.2.). By default, it uses the file name with an added "ds_" in front, but this can be changed as long as it starts with a letter and does not contain spaces or other characters not allowed in Matlab variable names. In this window, the sample interval (in seconds) of the measurement is also put in. Additional optional info about the dataset can be written in the last input field. This window is repeated for all spreadsheet files you have selected.

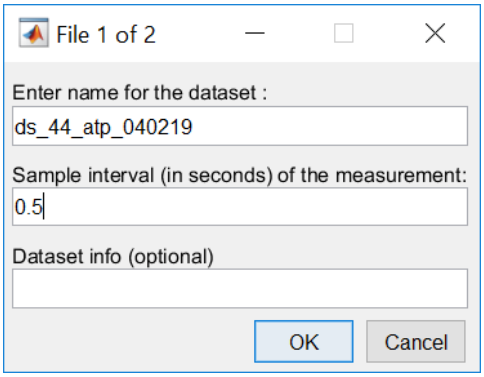


Figure 3.2 The file info input window.

After all the files are handled, the program asks whether the datasets should be added to an existing database or if a new one should be created. If you want to add the sets to an existing database,

choose that and locate the database file (.mat) in the selection window and the sets are saved there. **Note:** If the existing database already contains a dataset with the same name, it is overwritten!

If you would like to create a new database for the datasets, choose that option and give the filename you want (.mat ending is not needed). The database file is created in the same folder where the spreadsheet files that were read are located.

To view contents of a database, open it to the workspace by double clicking it in Matlab file browser. This loads a variable called *caDatabase* to the workspace. When opened, you can view and open all the datasets that the database contains (see more in Chapter 5).

Merging databases

In order to make statistical comparisons of your datasets (e.g. boxplot or scatter graphs, see Chapter 7), they should be in the same database (.mat file). So, when reading the spreadsheet files into Matlab, include all the datasets you want to compare in the same database. However, if you realise this later, databases can be merged with the merge tool.

First, make backups of the databases to be merged. Second, make sure that the databases do not contain datasets with same names, because they would be overwritten! Then, select "Merge databases" from the main menu and select all the .mat files you want to merge and give a name for the new merged database in the dialog box. If everything is successful, the program notifies you and returns to main menu. The merged .mat file of the new database is saved in the same folder as the files that you selected for the merge.

4. Performing the intensity analysis

Opening the data

First, you should have a database with imported datasets, as described in the previous chapter. Select *Analyze intensity data* from the main menu (Fig 3.1). Next, select the database file (.mat) and the dataset from the list dialog that opens after selecting the database. After this, the program asks which ROI and cell you want to start from. The ROI should always be specified, but the cell field can be left empty if you want to start from the first cell, which is not analysed (e.g. when returning to analysis of an incomplete dataset).

Bleach correction

Next, the intensity data of the cell that was chosen is plotted (Fig 4.1). In the figure, the blue curve is the original raw data and the green curve is the bleach corrected version. The part of the raw data curve that is in red is the baseline interval, which is used for fitting the baseline correction function. You can change this interval by answering *No* in the popup menu that asks if you accept the bleach correction and selecting the new interval start and end points with the mouse cursor. Only x-values are recorded, so simply just align the vertical line where you want the start or end point to be and press left mouse button.

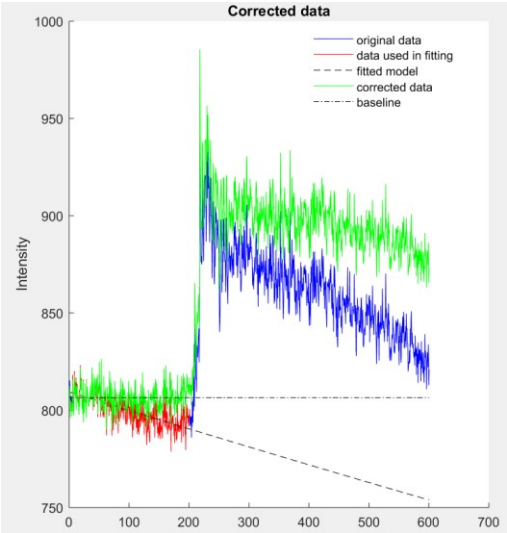


Figure 4.1 The bleach correction figure.

When you are satisfied with the bleach correction, answer *Yes* in the popup menu to continue to the actual analysis. You can also select *Skip correction* if you do not want to do bleach correction (analyse the original raw data) or *Discard response* if you want to discard the data altogether from the analysis (can still be plotted from the database later). If you want to go back and analyse a previous cell again, choose *Previous cell* or skip the analysis and go to the next cell by selecting *Next cell*.

Intensity analysis

When you accept the bleach correction, another graph is plotted to the right side of the first one. This is the bleach corrected version of the data, which is filtered to remove noise that would interfere with the analysis. In the graph are also reference lines for the baseline and halfway values and four calculated points, which are the following (Fig 4.2):

- **Response start (red)**: The time value where the intensity response starts. The program makes a starting guess for this. If the cell is the first analysed cell in the dataset, the guess is the end point of the baseline fit from the bleach correction. If there are already analysed responses in the dataset, the script calculates the average of their start points and uses it as a starting guess. If the guess is incorrect, the point can be moved (see later)
- **Response maximum (black)**: Simply the time value where the maximum intensity of the response occurs. It can be moved, if the interpretation of the maximum is incorrect (see later).
- **Halfway rise (green)**: The time value for when the intensity crosses the 50% threshold between the start and max values. The script looks for all the points where the threshold is crossed and picks a crossing from the middle (in case the intensity oscillates over the threshold). The value can be moved (see later)
- **Halfway decay (yellow)**: Similar to halfway rise, but after the maximum. Similar to the rise value, a middle threshold crossing is chosen, if the intensity oscillates over the threshold. The value can be moved (see later).

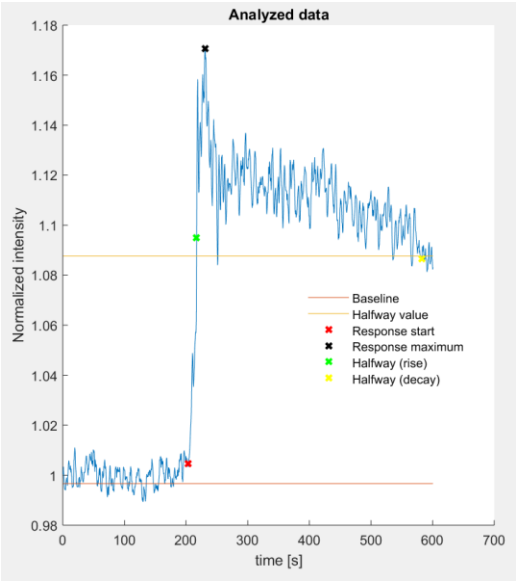


Figure 4.2 The response analysis figure.

If you are not satisfied with the calculated data points, they can be moved by selecting *No* in the popup menu asking if you accept the data. After this you can choose new data points with the cursor similarly to the bleach correction window.

Note: The points have to be clicked in order (start, halfway rise, maximum, halfway decay) and you always have to click four times. If you wish not to change a certain value, click with the right mouse button instead of left. For example, if you want to change only start and maximum points, click left (at the new start value), right (anywhere), left (at the new max value), right (anywhere).

Note: When the start and/or maximum values change, the script automatically calculates the new halfway value, so it is best to first adjust start and maximum and then check if the halfway points still need adjusting.

In the same menu, you can also choose to analyse Ca-sparking. If your data has sparking, choose *Add/Change spark data* and select the sparking interval with mouse cursor, similar to the bleach correction baseline. This plots the spark peaks to the data. You can also remove previously analysed spark data by selecting *Delete spark data*. When you are satisfied with the analysis, select *Yes* and proceed to next cell. You can return to the main menu any time from the analysis menu, the data is saved after each cell.

5. Handling databases

Data format

As noted in chapter 3, to view contents of a database, open it to the workspace by double clicking it (the .mat file) in Matlab file browser. This loads a variable called *caDatabase* to the workspace. When opened, you can view and open all the datasets that the database contains. If you open a dataset, a table is opened where you can see all the individual cells as *intensityResponse* objects. Each column corresponds to a single ROI. You can click open an object from the list to view its data. The data variables can be grouped as follows:

Dataset/cell info variables

datasetInfo: dataset info from user input

samplingInterval: the data sampling interval of the measurement (in seconds)

indices: a pair of indices to identify the ROI (first index) and the cell (second index) of the data

coordinates: the location of the cell center point in the original image (read from the spreadsheet) and the calculated radius of the circle (in pixels)

groupNumber: determined after the grouping algorithm is run, tells which group the cell belongs to

Data vectors

timeVector: the time vector read from the first column of the spreadsheet

rawData: the raw intensity data read from the columns of the spreadsheet

correctedData: bleach corrected version of the raw data

relativeData: normalized version of the bleach corrected data

filteredData: filtered version of the relative data

interpolatedData: [for possible implementation later]

baselineModel: the model calculated from the baseline fit and used in the bleach correction

Indices

fitIndices: start and end indices of the baseline fit

startmaxIndices: indices for the response start and max values

halfwayIndices: indices for the halfway rise and decay

sparkInterval: indices for the spark interval start and end

highSparkPeaks: indices for the spark peak maxima

lowSparkPeaks: indices for the spark peak minima

c*alcData*: [for possible later implementation]

Calculated data (see Fig 5.1)

maxAmplitude: relative amplitude of the response (response maximum/response start value)

rise50: time from response start to 50% intensity threshold before the maximum

time2max: time from response start to the maximum intensity

firstHalf: time from 50% threshold (rise) to maximum intensity

decay50: decay time from maximum intensity to 50% threshold

duration50: response duration between 50% threshold values

sparkTime: duration of the sparking interval

sparkStartTime: time when sparking starts

avgSparkInterval: average duration between the spark peaks

avgSparkAmplitude: average amplitude of the spark peaks

maxSparkAmplitude: maximum amplitude of the spark peaks

numberOfSparks: number of spark peaks in the spark interval

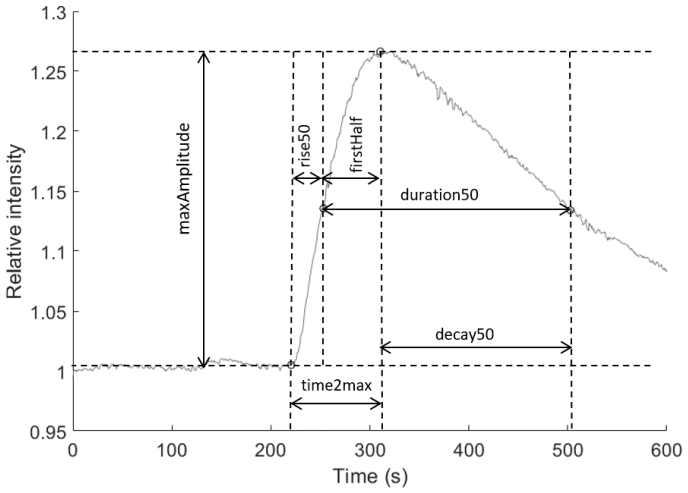


Figure 5.1 Schematic of the amplitude and time parameters calculated from the intensity response.

Logicals

isAnalyzed: changes to true when the response is analysed for the first time

isDiscarded: changes to true if the data is discarded (used for plotting only discarded data)

isSkipped: changes to true if the data is skipped

isSparking: changes to true if sparking is analyzed

Viewing data from a database

To view analysed data from an existing database, select *Browse or plot response curves* from the main menu and then select the database and dataset as before. A new menu pops up to ask which data you want to see (Fig 5.1).

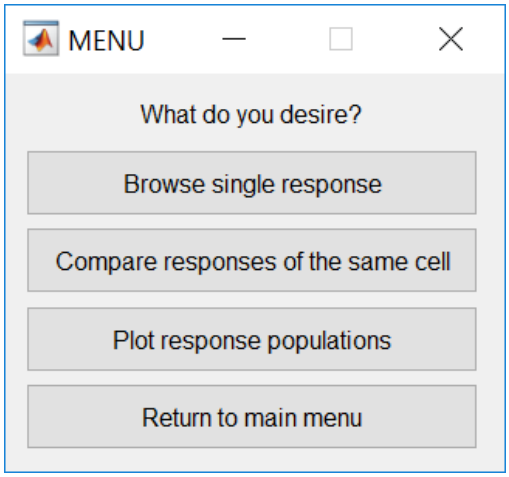


Figure 5.1 The data browsing menu

By selecting the first option, you can browse single responses one by one. Select starting ROI and cell in the selection window, and whether you want to view accepted or discarded cells. If the cell you have specified does not match your selection of accepted/discarded, the program finds the first one matching it, overriding your starting values.

If your data is arranged so that each ROI corresponds to the same cells in different measurements, you can compare their responses by selecting the second option in the menu. Select the starting cell in the input window, and the program plots the cell responses side by side (e.g. when starting from cell 1, it plots cell 1 of ROI 1, cell 1 of ROI 2, etc.). Before plotting, the program asks you to enter names of the ROIs (e.g. ATP1, ATP2, etc.). Separate the names with spaces and make sure the amount matches the amount of the ROIs.

The third option in the menu lets you plot response populations in the same figure. When selected, you are asked to specify if you want all the responses to the same figure, or separated by ROIs. Either way, the program plots both accepted and discarded (if any) responses and calculates their amounts.

6. Cluster analysis

Cluster analysis uses a grouping algorithm that classifies the responses according to the calculated parameters given by the intensity analysis. Thus, it forms groups of similar responses from the dataset and visualises them.

Working principle

For now, the clustering algorithm uses the following calculated parameters from the analysis to perform the response grouping: *maxAmplitude, rise50, time2max, firstHalf, decay50* and *duration50*. All parameters are normalized by the amplitude, to root out the effect of larger responses having larger time parameters, and to find the different shapes of curves as well as possible. Discarded

responses and outliers (values more than 3 standard deviations away from the mean) are left out of the grouping.

With this data, the algorithm first uses principal component analysis (PCA) to calulate new parameters, which explain most of the variance in the data and thus create the best clusters. PCA shifts the data into a new coordinate space that has the calculated principal components as axes. The clustering info figure (see below) plots the data in the new coordinate spaces to visualize the clusters.

The actual clustering is done with original Matlab functions, such as *linkage* and *cluster* that create a hierarchical cluster tree from the data and calculate the optimal number of clusters (currently between 1 to 4) so that they would be as unique as possible. When the clustering is complete, the original responses are assigned to groups corresponding the clusters and each groups responses are plotted. The clustering info figure (see below) displays information on how the groups differ from each other.

Using the cluster analysis tool

Select *Cluster analysis* from the main menu and a dataset that has been already analysed. **Note:** All cells of the dataset have to be analysed in order for the cluster analysis to work!

When a dataset is selected for analysis, the program opens two figures for each ROI: a 2x4 figure of analytical information of the clustering for the ROI and the actual intensity responses grouped into their corresponding groups.

Clustering info figure

The first plot of the info figure (Fig 6.1) is a pareto chart of the principal component analysis (PCA). It displays the best principal components of the data (bars) and a line showing the cumulative amount of how much variance of the data the principal components explain.

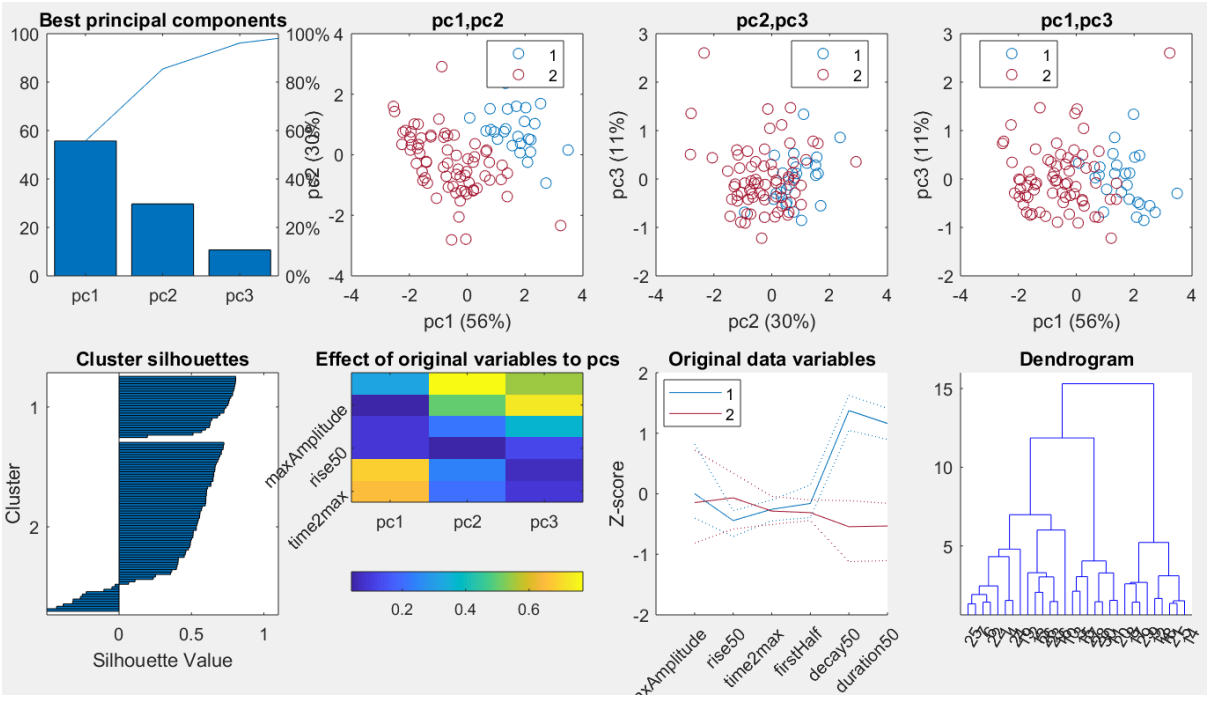


Figure 6.1The cluster analysis info figure

The next three plots in the upper row visualise the data in the coordinates defined by the new principal components. The data points are color coded regarding their groups to see how the data forms clusters in the new coordinate space.

The leftmost plot in the lower row visualises the silhouette value of each group and data point. The silhouette value for each data point tells how well it matches to other data points in its own cluster. It ranges from -1 to 1 and higher silhouette value means a better match. The grouping algorithm tries to form a number of groups, which minimises the amount of data points with negative silhouette values.

Second plot in the lower row shows how much each original variable affects each principal component. Next to it is a plot that is probably most useful for interpreting the grouping. It shows a line chart for each group and how their grouping parameters differ from each other. The values are given as Z-scores (average subtracted from the value, divided by the standard deviation) to make them comparable. Therefore, this figure gives you a quick glimpse of how the individual groups are different and what kind of responses they contain.

The last plot in lower right is a dendrogram, which shows how the grouping algorithm forms the main groups by combining smaller subgroups.

The clustering info figure for each ROI is saved in the current folder with the name *<dataset name>_clusterinfo_ROI<number>.fig.*

Response grouping figure

This figure is opened alongside with the analytics figure for each ROI of the dataset (Fig 6.2). It presents all the groups formed by the clustering algorithm and the number of responses in each group (in the title).

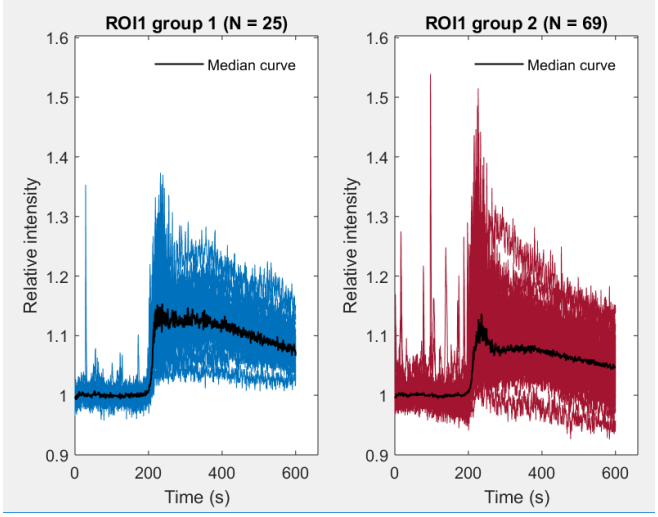


Figure 6.2 The response grouping figure

The black curve in the middle of the response population is the median curve calculated from all the responses in the group. This can be copied for example to a summary figure, presenting all the median curves of the ROI. The grouping figure for each ROI is saved in the current folder with the name *<dataset name>_grouping_ROI<number>.fig.*

7. Visualising the data

Visualising the grouping and location data on the original image

Once the cluster analysis has been run for a given dataset, its results can be visualized back to the original image (Fig 7.1). It is recommended to use for example an image of averaged intensity from the whole measurement (Z-project in ImageJ), or a single frame if that serves the purpose.

Select *Location Data* from the main menu and open the database and dataset you want to analyse. The program asks the pixel/length ratio of the original image in order to place the markers correctly. You can check the length of the image side in pixels and micrometers from ImageJ. So far, the script only supports square ROIs. For example, if you have a 200x200 pixel image with a side length of 104 µm, write 200/104 in the input field (the script calculates the value itself). Next, select all the image (.jpg) files of the dataset (all ROIs), and the program draws an overlay of color-coded markers on all the ROI images next to each other (this can take a while if there are many cells and ROIs).

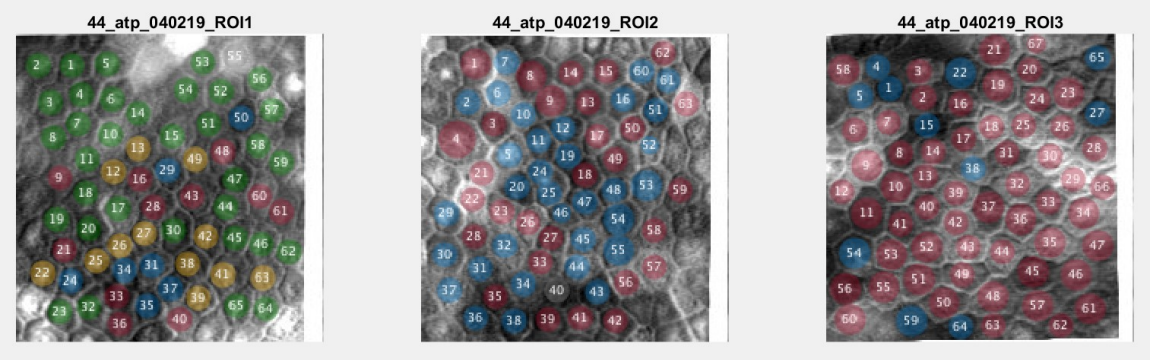


Figure 7.1 The data location overlay images

The groups are color-coded with the same colors as in the grouping figure on the cluster analysis tool. This makes it easy to compare how each group's responses are located in the image field. The responses are also numbered, so each response can be looked up with the response browsing tool, or have its numerical data checked from the database. Each overlay image is saved as a .jpg file to the same folder as the original with the name *<original name>_edited.jpg*.

Boxplot visualisations

The calculated data variables of each dataset can also be visualized with different methods within the program. The first one of these is a boxplot visualization (Fig 7.2). Select *Statistical visualizations* from the main menu, and open the database and dataset in question. The program then asks which datasets you want to include (multiple can be selected). If only one dataset is selected, you can further select which ROIs are plotted.

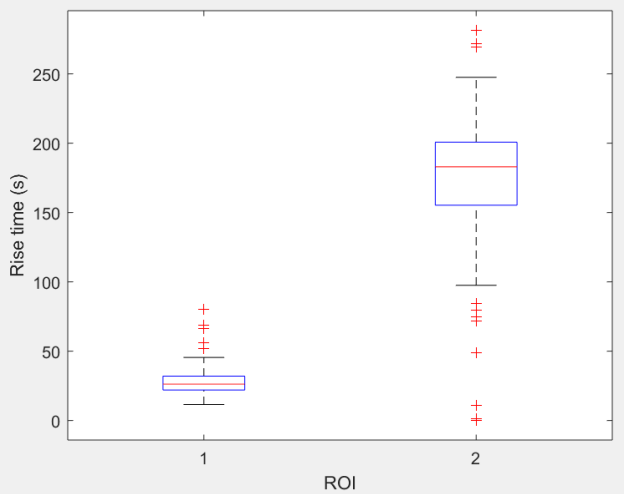


Figure 7.2 Boxplot visualization of rise time data for two ROIs

The program also saves the data used in the boxplot as a spreadsheet to the current folder with the name *boxplotdata_<dataset name>.xlsx* for later editing in other programs. The ROI or dataset numbers are presented in the second column.

Scatter visualisations

To make a scatter plot, open the database and dataset similar to the boxplot visualization and select *Scatter plot* in the plot selection menu. Next, select variables to plot in x- and y-axis, respectively. The program now draws a scatter plot (Fig 7.3) using data from the selected ROIs with the same color coding used in the cluster analysis.

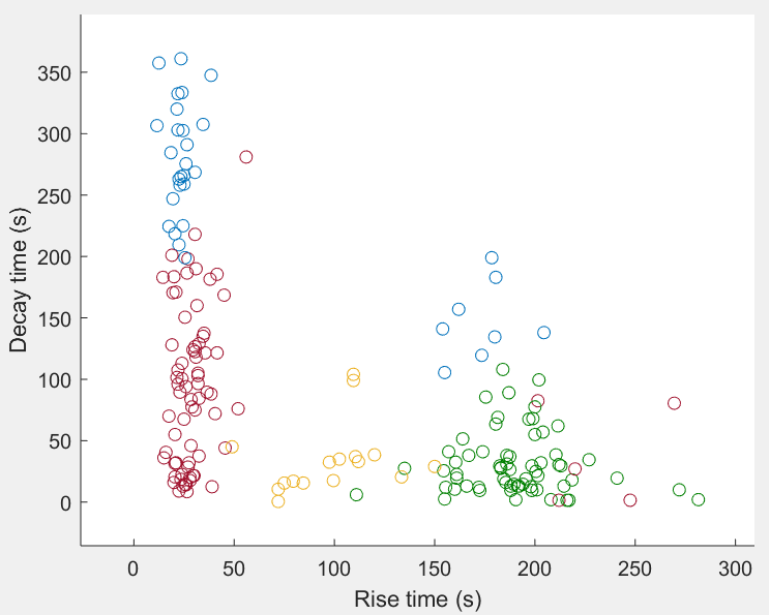


Figure 7.3 Scatter visualization of rise and decay times of two ROIs

The scatter plot data is also saved in the current folder as a spreadsheet with the name *scatterdata_<dataset name>.xlsx.* The first column is the x-data, the second is the y-data and the

third is the group number. Group number 0 means a discarded response (not visible in the scatter plot).

Note: Be careful when combining ROIs in the same plot. This can result in split groups as in Figure 7.3. The blue group is split in two populations arising from group 1 of ROI1 and group 1 of ROI2. It is best to use the scatter plot with a single ROI at a time, unless the groups are very similar in multiple ROIs, or at least plot the scatters first separately and identify which populations are from which ROI.

8. Basic workflow

Below is a flowchart presenting the basic workflow from beginning to end of analysing Ca-imaging data with the program. This takes into account that some stages of the analysis have to be completed before others.