

Manual: S3T



Manual: S3T-based analysis of synaptic imaging experiments

Table of Contents

1	<i>Objective</i>	5
2	Keywords	5
3	Source	5
4	Overview.....	6
5	Procedures	8
5.1	Prerequisite	8
5.1.1	Installation of S3T.....	8
5.1.2	Input.....	9
5.2	Image processing	10
5.2.1	Start the program.....	10
5.2.2	Load and process image files.....	12
5.2.3	Info during Processing	12
5.2.4	Output.....	13
5.2.5	Post processing:	13
5.2.6	Data analysis and reporting:	14
6	Appendix:.....	15
6.1	Parallel processing:	15
6.2	Experimental features:.....	15
6.2.1	Program Buttons.....	16
1.a.1.1.	Analysis configuration fields	16
1.a.1.2.	Fast Load.....	16
6.3	Threshold Settings.....	17
6.3.1	Options button (legacy features).....	17
1.a.1.3.	Load Buttons.....	17
1.a.1.4.	Play Buttons	17
1.a.1.5.	Image mask creation buttons.....	18
1.a.1.6.	Signal processing buttons	18
6.4	Tool Buttons	18
6.4.1	Plate Layout	19
6.5	Analysis Tools	21
6.5.1	Data viewer.....	21
6.5.2	Trace/plate Viewer:.....	27
6.5.3	Project viewer:	33
6.5.4	Database Viewer:.....	37
6.5.5	Tif Manipulator:	37

6.5.6	plate Meta Data Converter	40
6.5.7	Analysis configuration file(s).....	40
6.5.8	Analysis configuration Tool.....	40
6.6	Files	44
6.6.1	Setup Files	44
1.a.1.7.	urls.json	44
6.6.2	Input Files.....	45
1.a.1.8.	barcode.txt.....	45
1.a.1.9.	*_Analysis.xml	45
6.6.3	Output Files.....	46
1.a.1.10.	*_analysis.txt	46
1.a.1.11.	*_traces.csv	47
1.a.1.12.	*_PPsynapses.txt	47
1.a.1.13.	*_RawSynTraces.csv	49
1.a.1.14.	*_synapses.txt	49
1.a.1.15.	*_synTraces.csv	50
6.6.4	Log Files	50
1.a.1.16.	Process_*.tif.txt	50
1.a.1.17.	processSoft_*.zip	51
6.7	Multi experiment batch processing.....	52
6.7.1	File processing in multiple directories	52
6.7.2	Processing a directory with *.tif files	52
6.7.3	Loading Analysis.xml	53
1.a.1.18.	Processing analysis files	53
6.8	Movie processing	54
6.8.1	Overview	54
6.8.2	Frame selection	55
6.8.3	Mask	55
1.a.1.19.	Mask loading	55
1.a.1.20.	Mask creation	55
6.8.4	Signal processing	56
6.8.5	Mask creation details	57
1.a.1.21.	Singular Value Decomposition (SVD)	57
1.a.1.22.	Thresholding.....	58
1.a.1.23.	Island detection and numbering	58
1.a.1.24.	Export MaskBTT	58
6.8.6	Signal Processing details	59
1.a.1.25.	Background subtraction	59
1.a.1.26.	Signal extraction	59
1.a.1.27.	Export synapse signals.....	59
1.a.1.28.	Export Mask	59

1.a.1.29.	Partial Processing (PP)	59
1.a.1.30.	Analyze single synapse signals.....	59
1.a.1.31.	Average Synapse Response	60
1.a.1.32.	Temporal averaging.....	60
1.a.1.33.	Analyze average Response	60
6.9	Algorithm details	60
6.9.1	DFF Calculation.....	60
1.a.1.34.	Linear Bleach correction	60
1.a.1.35.	2xExponetial Bleach correction.....	60
1.a.1.36.	Auto2expInt correction	61
1.a.1.37.	Autolinint correction	61
1.a.1.38.	Global correction	61
6.9.2	MR1 (Multi Response to 1)	61
6.10	Installation	62
6.11	Errors and bad analysis	62
6.11.1	Errors during processing	62
6.11.2	Bad analysis	62
7	References	64

1 *Objective*

Neurons communicate with other target neurons in a precisely timed and well-coordinated process called synaptic transmission. The key steps in chemical synaptic transmission are: action potential generation in presynaptic neurons, neurotransmitter release from presynaptic terminals and integration of synaptic inputs to the postsynaptic neurons. Measurement and quantification of each of these steps is essential to understand the information processing in the brain. To this end, we have developed a desktop analysis software (S3T) which extracts and quantifies the responses of the stimulated synapses. This document describes how the data generated with the high-throughput imaging are analyzed.

2 **Keywords**

S3T; Image analysis; Synaptic Imaging

3 **Source**

Name	Vendor	Cat No	Storage
S3T (ver. 0.91)	Cell Histology Lab - Universiteit Antwerpen (M. Van Dyck) (https://github.com/meChiel/s3T)		Local analysis computers, NAS, S3

4 Overview

There are two parts of HTS imaging:

1. From cells to imaging
2. From image files to data analysis

Current document which describes how image files (tif) are analyzed with imaging software-S3T and further reported. The image files are transferred from the imaging PC to the NAS, where the data is analyzed, reported and subsequently moved to Amazon storage (S3).

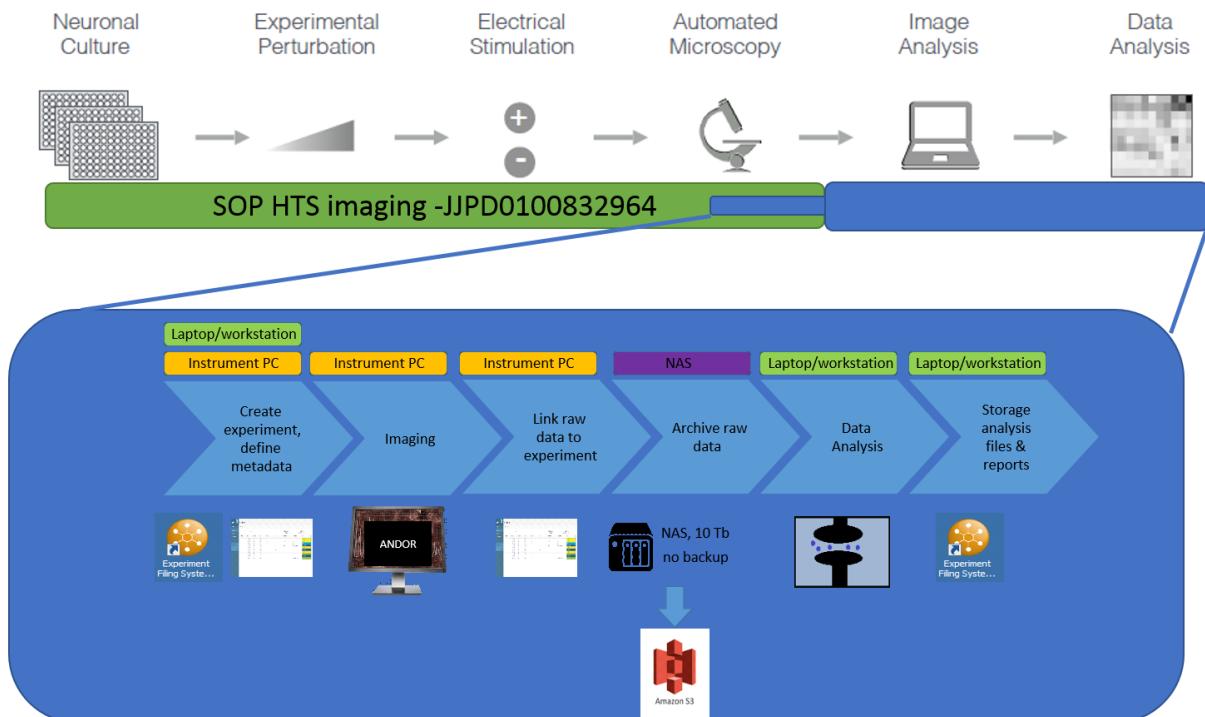


Figure 1. ...

A typical experiment starts with the 96 well plate of which only the inner 60 wells are seeded with iPSC-derived or primary hippocampal/cortical neurons. This results in a data set of 60 single channel fluorescence time lapse movies. Typically, the frame rate used for the acquisition is 30 frames/second (fps). In the movies the signal fluctuations of the synaptic markers in response to electrical stimuli are captured. This document describes the analysis of the synapse signals.

Image analysis

In brief, per movie, the intensity fluctuations are measured at the individual synapse level. For this, one or more masks are applied, in which the signal change is measured as a function of time. The signal kinetics (traces) are in turn modelled and processed to remove undesired effects and to extract descriptors. For all image stacks, the traces and descriptors are stored in data files, which can be analyzed and plotted for specific treatments (compounds).

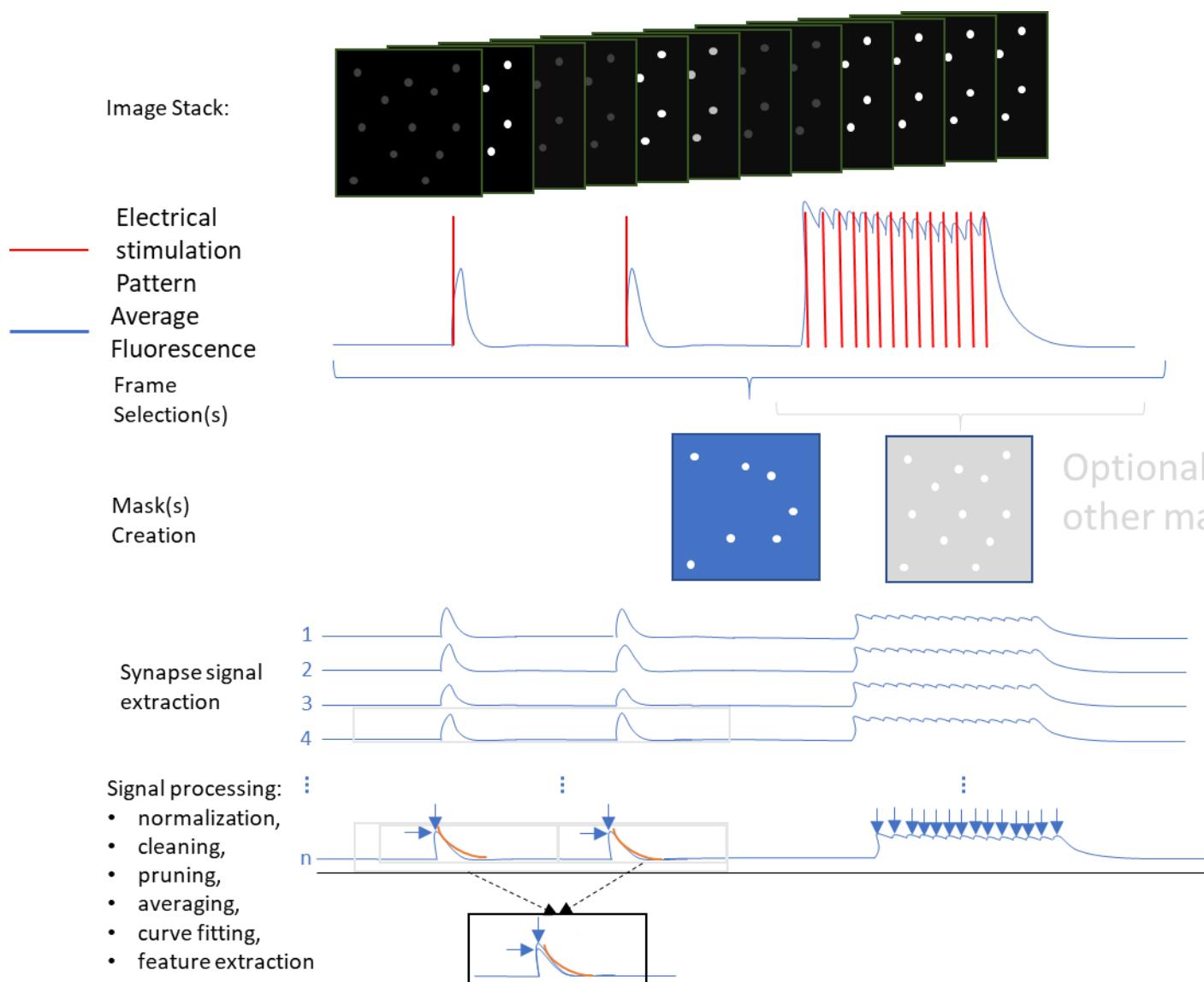


Figure 2....

The S3T software returns the information on synapse dynamics (traces and descriptors) in *.csv files. For a more integrated data exploration or secondary analysis, with functions and overviews tailored towards insight into the synapse data, the experimenter/user can use the data viewer application (built in feature in S3T) or other software such as Excel, GraphPad, Igor Pro, etc. In the data viewer, the user can explore descriptors at different levels of the experiment ranging from the experimental setup where typical dose response curves can be visualized, over the image and down to single synapse level, where the response of a single synapse can be analyzed. This to allow traceability throughout the analytical workflow and to offer facile means for curation (for detail see Appendix 6.5 Dataviewer).

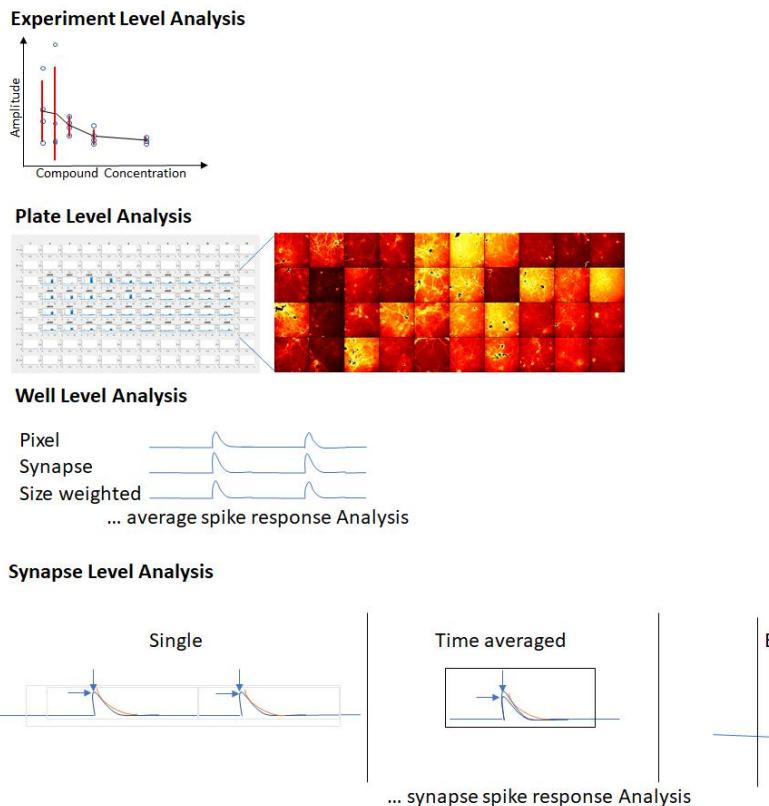


Figure 3: Different analysis levels for which statistics are generated with S3T and for which the data can be interactively explored with the S3T:dataviewer tool.

5 Procedures

5.1 Prerequisite

5.1.1 Installation of S3T

Check and install the latest version of S3T software from

Or <https://github.com/meChiel/S3T/releases>

A separate urls.txt file should be placed in the c:\Windows\urls\ dir which contains the location of the PLTS server. This allows to automatically download the plate meta data (for detail see Appendix: plate Meta Data Converter).

5.1.2 Input

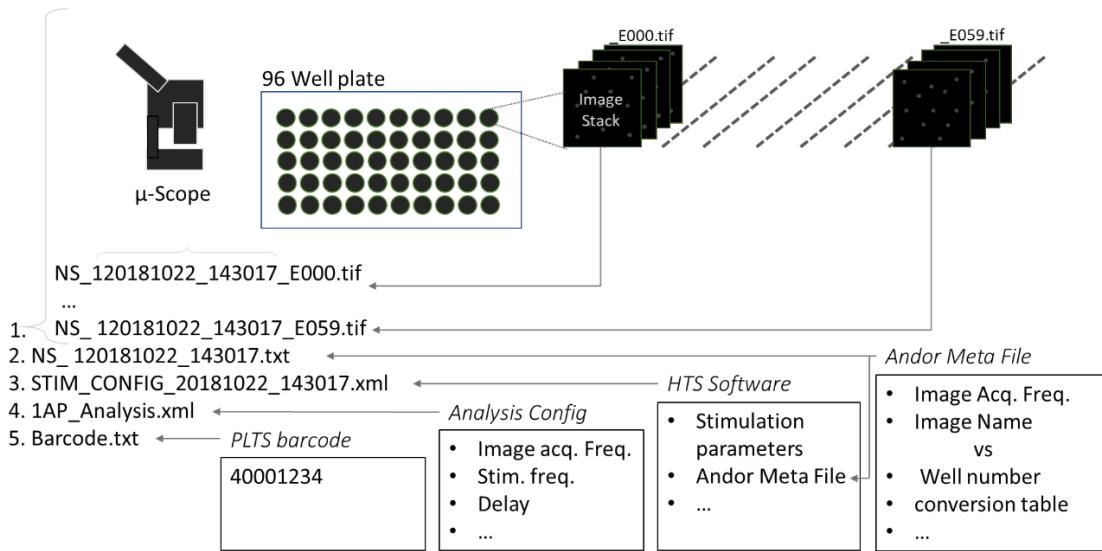


Figure 4: Different types of data files which are brought together in the processing pipeline.

There are several steps or files needed to proceed before HTS image files are analyzed.

1. **HTS Imaging files (tif files):** Files to be analyzed. (typical example - **NS_120181022_143017_e0000.tif**), where NS stands for neuroscience assigned by the experimenter, 1- number of experiments on a particular date - assigned by Andor IQ3, 20181022- date of experiments - assigned by Andor IQ3, 143017- time of experiments - assigned by Andor IQ3, e0000 - file number always start with 0 - assigned by Andor IQ3. Typical dimensions are 512 x 512 pixels, recorded with 40x objective, 33fps, and somewhere between 200 and 3000 frames.
2. **Andor meta data file:** This file is generated automatically by the acquisition software (Andor IQ3) and saved in the folder which contains HTS imaging files. This text file (ex. **NS_120181022_140908.txt** - nomenclature same as *.tif files explained in previous section) contains information about microscope settings, camera settings, well information (well number and x, y, z coordinates) - for details see SOP-HTS imaging (JJPD0100832964).
3. **HTS (electrical stimulation) meta data file:** (ex. **STIM_CONFIG_20181022_143017.xml**) which contains the information about the stimulation protocol, generated by HTS software and stored in the folder where HTS imaging files are located - for details see SOP-HTS imaging (JJPD0100832964).
4. **Analysis configuration file(s):** File(s) which define/configure the type of analysis which should be done on the synapse fluorescent response data. This file contains the acquisition frequency, time window of interest. Synapse Mask creation settings (for detail see Appendix: 6.5.7 analysis configuration file). When the analysis is not available, you can create a new analysis file with Analysis configuration tool or copy and modify the *_Analysis.xml files from a previous project.
5. **Barcode.txt** - contains barcode generated in PLTS which contains the plate layout information (for detail see Appendix: 6.5.6 plate Meta Data Converter).

5.2 Image processing

5.2.1 Start the program

Find the icon of the S3T program and run it as administrator.

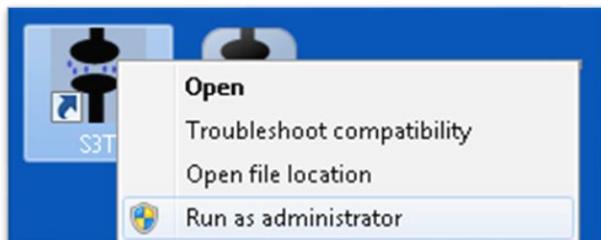


Figure...

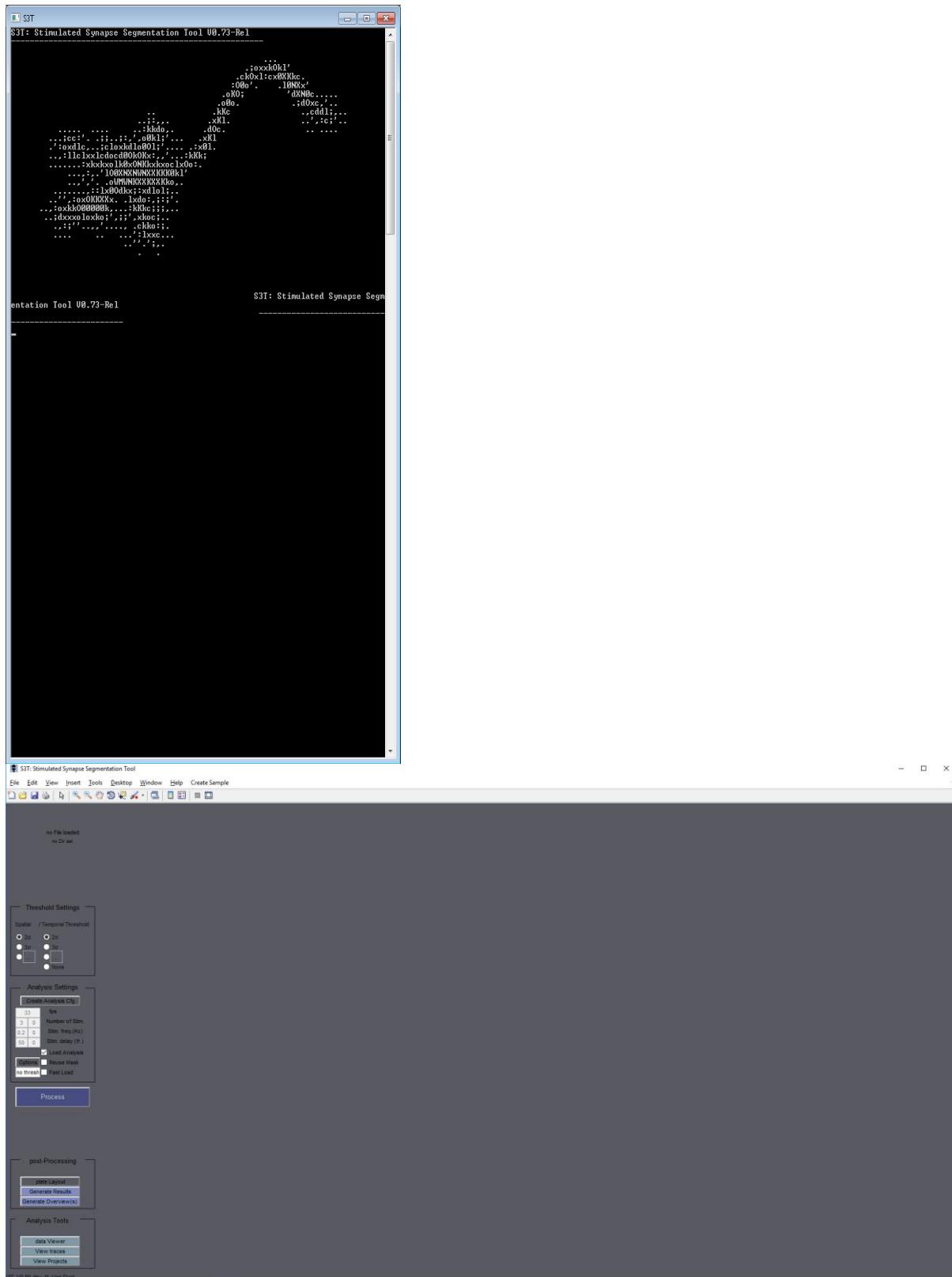


Figure1: S3T startup screen

A command line window of the software will open (see figure on the left).

A GUI for the software will open as well. (See figure right).

5.2.2 Load and process image files



Process :

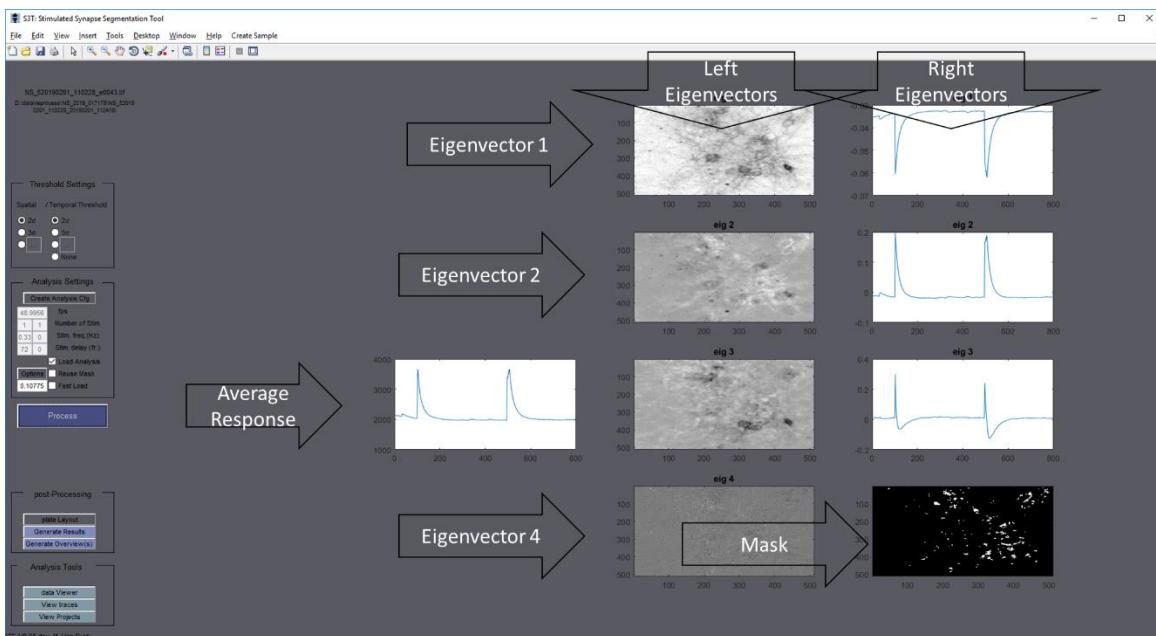
The process directories button opens first a window where you can specify the directory, then it will process all the *.tif files in that directory and all the movies found in the subdirectories. The type of analysis is defined by the one or more analysis files which are copied into the data directory before pushing this button. If load analysis is disabled, the user can input analysis parameters (1.a.1.1 Analysis configuration fields) or can make a custom analysis with the (6.5.7) analysis configurator. To speed up the processing multiple instances of S3T can be run simultaneously (for details see Appendix: 6.1 Parallel processing)

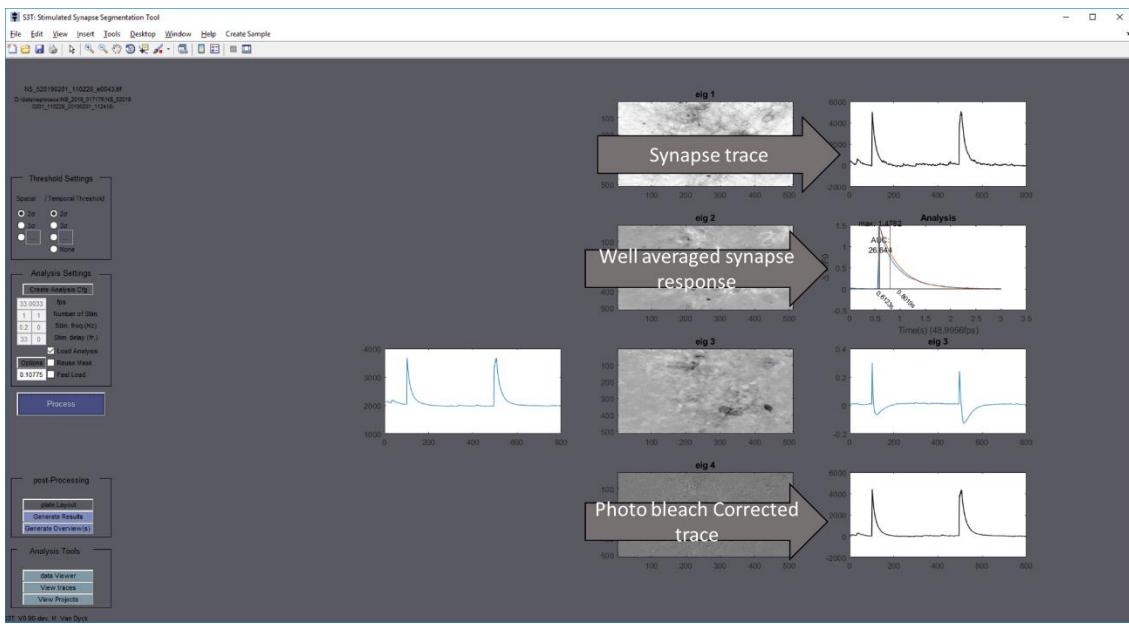
Options: This button gives access to experimental features of the software and should only be used for experimentation. Some documentation of these features is given in the appendix 6.2 : Experimental features.

5.2.3 Info during Processing

Top Left Text: In the top left of the GUI, the currently processed file/ directory is displayed. When all files in a directory are processed the text displays: *All files in ... are processed*.

Images: During the processing of the movies, the different windows show the different plots and sub-results which are generated. This allows you to get direct feedback if the analysis is done correctly. Plots which are generated are: Time average, Analysis, signalPlot, mask, all pixel average; All synapse average. Aligned responses. Also, the first 3 singular value decomposition (SVD's 1.a.1.21) vectors of the movie (left and right eigenvalues) are displayed. For details see Appendix 6.6 Files





Command window: In the command window, errors, warnings, and debug results are shown during processing.

Process folder: Any error during processing of a tif file can be read in the command window. To retry processing of that file delete the .txt file with similar name in the Process folder. Otherwise, a rerun will skip this movie from processing again (for details see Appendix 6.11 Errors).

5.2.4 Output

6 different files with distinct features/information are exported for each .tif file after analysis.

For each analysis (*.Analysis.xml) a subdirectory is made with the same name. This contains an output directory with two types of files:

1. The *.analysis.txt file contains several features for a particular well/tif file. Typical features are amplitude, frame when the maximum amplitude was reached, decay constant, Area under the curve, ... (for more details see Appendix- 6.6.3 Output files)
2. The other *.trace.csv file contains different types of averaged time traces. E.g. pixel average, synapse average, synapse size averaged trace (for more details see Appendix- 1.a.1.11).



NS_920180820_1547NS_920180820_1547
09_e0045_PPsynapse09_e0045_synapses.

In the synapse details subfolder, 4 other files are generated:

1. The *_syntraces.csv file contains the ΔF/F trace for each individual synapse. (See Appendix 1.a.1.15)
2. The *_Synapse.txt file contains all the features for each individual synapse. (See Appendix 1.a.1.14)
3. The *_PPsynapse.txt file contains multiple features for each individual subsection of the trace. Typically used for burst analysis. (See Appendix 1.a.1.12)
4. *_RawSynTraces.csv: file contains the trace before ΔF/F calculation. Black level subtraction is done but no background correction. (See Appendix 1.a.1.13)

5.2.5 Post processing:

When the analysis for all the movies is done, some summary files can be generated:

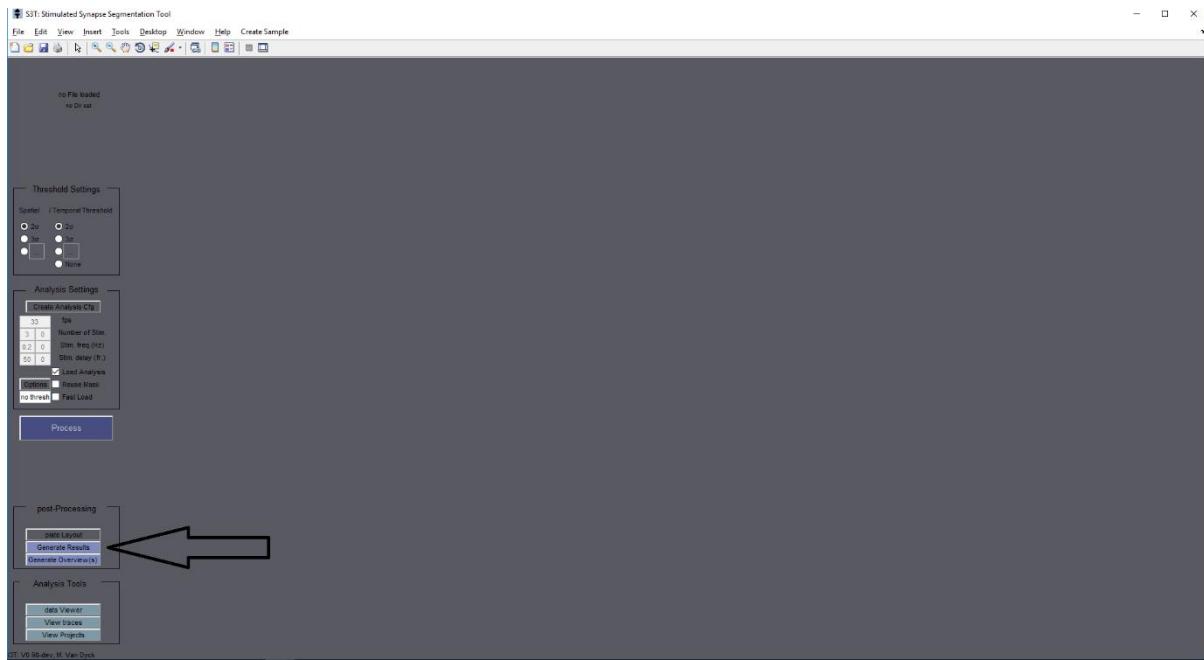
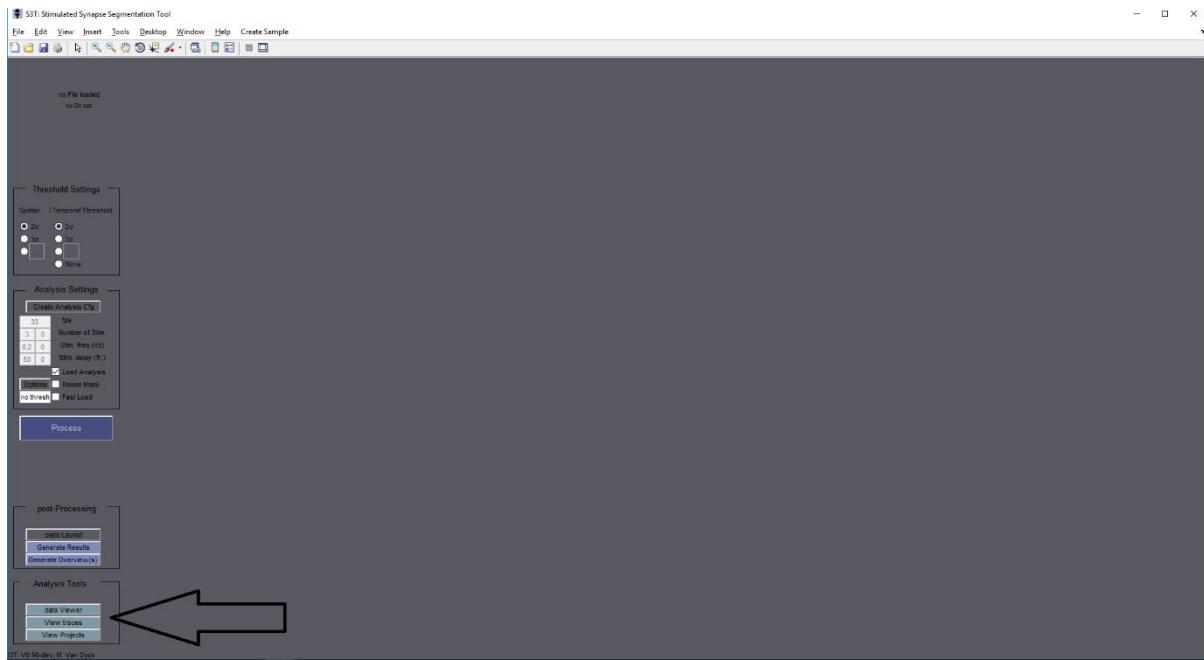


Figure....

- **Generate results:** This will create for each analysis an **AllWells.txt** file containing a concatenation of the analysis of each well in a 96-well plate. It will also concatenate the features extracted from the **PLT server** to each well. You should specify a directory: The barcode in the barcode.txt files in all sub-directories will be retrieved and will generate the **platelayout_COMPOUND.csv** files. This data is then used to concatenate the meta-data on the analysis results (for detail see Appendix:6.4.1.1.1 and 6.5.6 plate Meta Data Converter).
- **Overview generator:** This creates a collage (a plate layout) of all the .png file of a particular type; e.g. mask, analysis, signalPlots, temp. Averaged, Spatially averaged, eigenvectors... (see Appendix 6.4.1.1.2)

5.2.6 Data analysis and reporting:

Using the created AllWells.txt file, the different conditions (compounds, controls, etc) and parameters (peak amplitude, area under the curve; AUC, decay time constant) can be plotted in GraphPad Prism 7, excel, or with any of the S3T: Analysis Tools: S³T Data Viewer, S³T Trace viewer, S³T Project Viewer or any other software.



6 Appendix:

6.1 Parallel processing:

Processing of the files can be accelerated by using more cores/computers/clusters/cloud infrastructure.

Multi-core: A lot of the processing especially the calculation of the SVD is done in parallel over the different cores in a multi core machine. Matlab does not use hyper-threading. So, at most 50% of the machine is processing. Starting up a 2nd S3T instance of the program will increase the usage of the processors. A lot of the processing time is used by loading the movie into memory. This does not require any processing power. Also, the writing of the results and some intermediate algorithms do not use a lot of compute power, for this, 3 or 4 instances can run on a machine depending on the number of cores in the machine but also and more restricting is the RAM usage which limits the number of S3T instances you can run on 1 machine.

Compute-Cluster: You can also accelerate the processing by running the program on multiple computers. The program will automatically distribute the work over the different computers. You do not need MPI or another job scheduling/distribution system.

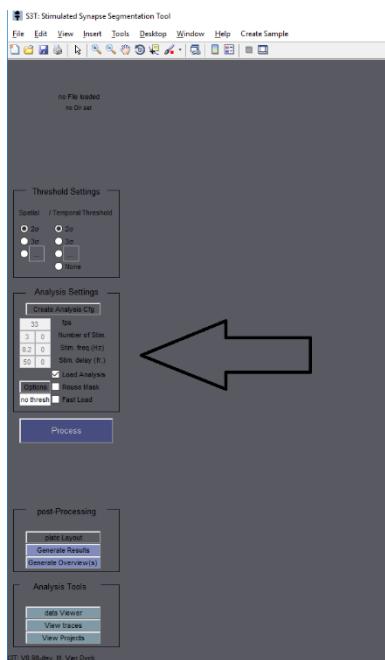
AWS: EC2 (Amazon web services: Elastic Cloud Computing): S3T can also run on amazon computers in the cloud. Here multiple computers can be rented. To speed up processing dramatically. Downside is that the data has to transferred to the cloud first, which takes some considerable time.

6.2 Experimental features:

A quick overview of experimental features / undocumented / unsupported / no warranty buttons: These should only be used for exploration and debugging. The documentation of these features can become obsolete.

6.2.1 Program Buttons

1.a.1.1. Analysis configuration fields



fps field: The framerate of the movie (i.e. the number of microscope images acquired per second) should be set in the fps field.

Number of Stim: The number of electrical stimulations per second should be entered in the Number of Stim. Freq. Field. (e.g. 0.125 Hz means 1 stimulation every 8 seconds.)

Stim. delay: Stimulation delay is the number of frames that should be skipped from the beginning of the movie.

The **Load Analysis** checkbox (default on) defines if the software should look in each directory for an analysis file containing these values and load these in the fields before processing the movies in that directory. The name of this file should be "***_Analysis.xml**" which contains the values for all the previously mentioned fields. If a certain file in the directory should be processed differently e.g. the file PSD-GCaMP6f_1.tif should be processed differently. Then a file called

"PSDGCaMP6f_1.tif_*_Analysis.xml" can be created with the values for analysis stored in, for that particular movie. This particular movie is not processed with the directory default analysis scripts. Multiple analysis scripts can be stored in a directory. The result of the analysis is stored in a subdir named after the analysis file. i.e. the `./*_Analysis/` subdir and the **PSDGCaMP6f_1.tif_*_Analysis** subdir.

reuse mask: The reuse mask checkbox indicates if a previously generated mask should be reused instead of calculating a new one. Only check this box if a previously calculated mask is present or a mask is copied from another movie. The mask is a special *.png file stored next to the tiff file. So for name.tif the mask is stored in "name.tif_mask.png". The *.png file is a 16bit png file with the background index 0 and the synapses index 2^{15} -synapseNumber. This file format allows identifying the different synapses afterwards. 2^{15} different synapses can be stored in this file. Each pixel can only be part of 1 synapse.

1.a.1.2. Fast Load

Fast Load: The fast load checkbox loads a low dimensional reconstruction of the movie instead of the full movie. This low dimensional reconstruction is based on the Singular Value Decomposition (SVD)/ Principle Component Analysis (PCA) which is computed when processing the tif movies. Currently the 16 most important left and right eigenvectors with eigenvalues are stored. These components are stored in the `./eigs/` subdir. Typically, the loading of the movies is accelerated by a factor x100. Depending on the content of the movie this reconstruction is less or more accurate. For electronically stimulated synapses, these reconstructions are fairly accurate. For unsynchronous spontaneous activity these reconstructions might be very different than the original.

6.3 Threshold Settings

- **Spatial Threshold:** allows to select the sigma of the Otsu threshold algorithm.
 - **Other** allows to set a fixed threshold.
- **Temporal Threshold:** will verify that for each synapse the response of the identified synapse is at least bigger than 2 or 3 sigma with sigma the std of the time-trace of that synapse.
 - **Other** allows to specify a certain $\Delta F/F$ value the synapse must cross. If not, the synapse is not used when calculating the average signal.
 - **None:** does not remove synapses based on certain threshold criteria.

6.3.1 Options button (legacy features)



1.a.1.3. Load Buttons

File: This allows to load a single *.tif movie instead of a full directory.

Dir: The Dir button allows to set the current working directory/The directory where the movies are stored. When pushing the process button, this directory will be processed then.

Project: An almost obsolete button to load a project file. This was a file where different directories could be defined accompanied with their properties. E.g. stimulus, time after addition of compound, ...

1.a.1.4. Play Buttons

Play: Will play the loaded movie. Typically, the framerate is quite low. To accelerate the playback speed, you can resize or click on a smaller window.

ImageJ: When lucky, ImageJ opens the current movie in ImageJ.

Slider 1: allows to traverse through the different frames of the movie.

Slider 2: nothing

1.a.1.5. Image mask creation buttons

Segment: The currently loaded movie is segmented based on the SVD (1.a.1.21). The value next to this button defines which eigenvalue is used to create the mask.

Segment cell bodies: This checkbox modifies the mask generation tool to find cell bodies instead of synapses.

Max z, mean Z, change are different ways to create a mask, First and second are by making a max or mean projection and the third is by calculating the absolute difference with the mean for each frame and do a mean projection again.

3sig, 2sig calculates the 2 or 3 sigma standard deviation value of the mask image. This value is shown in the field next to threshold button.

Threshold: This button will threshold the mask file: convert a grayscale image to a binary image.

2d freq., Tophat: 2 methods to suppress very big (e.g. spatial gradients) and very small (e.g. noise) features in the mask.

Clean: Does a morphological open operation to remove small objects in the mask.

rmvBkG: Does a threshold and clean operation in 1 click.

Detect I..: detect islands will convert the binary image to a segmentation where each individual island is labeled with a unique number.

Save ROI: makes a separate ROI *.png file for each individual synapse in the image.

Exp. Mask: export mask creates one special *_mask.png file with each synapse labelled with a different index.

Load mask: shows a popup screen to locate a mask file to be loaded.

1.a.1.6. Signal processing buttons

Extract ...: extract signals will acquire the average temporal behavior of each synapse out of the movie.

signalPlot: Plots for all the synapses the \delta f / f trace vertically spaced by 1.

HeatMe: shows a heatmap of all the synapses.

Avg Syn..: calculates the average response of all synapses.

Fold spi...: Fold spike response: will average repeated synapse responses according to the analysis parameters given.

Analyze: will fit an exponential to the response of each individual synapse

Analyze will fit an exponential to the averaged synapse response.

ExportS: will export a file containing the temporal behavior of each individual synapse.

ExportS: will export a file containing the dynamic response of the average of all these synapses.

Dose R: (obsolete) calculates a dose response curve based on project. Files and directories should be ordered in a certain way.

Peakfinder: is an experimental feature.

Load experiment: (obsolete) you can define a project file and it will process all related movies

Batch experiment: (obsolete)

Load Sti.: loads the parameters found in the stimulation.xml file in the current directory in the analysis fields on top of the S3 GUI.

6.4 Tool Buttons

6.4.1 Plate Layout

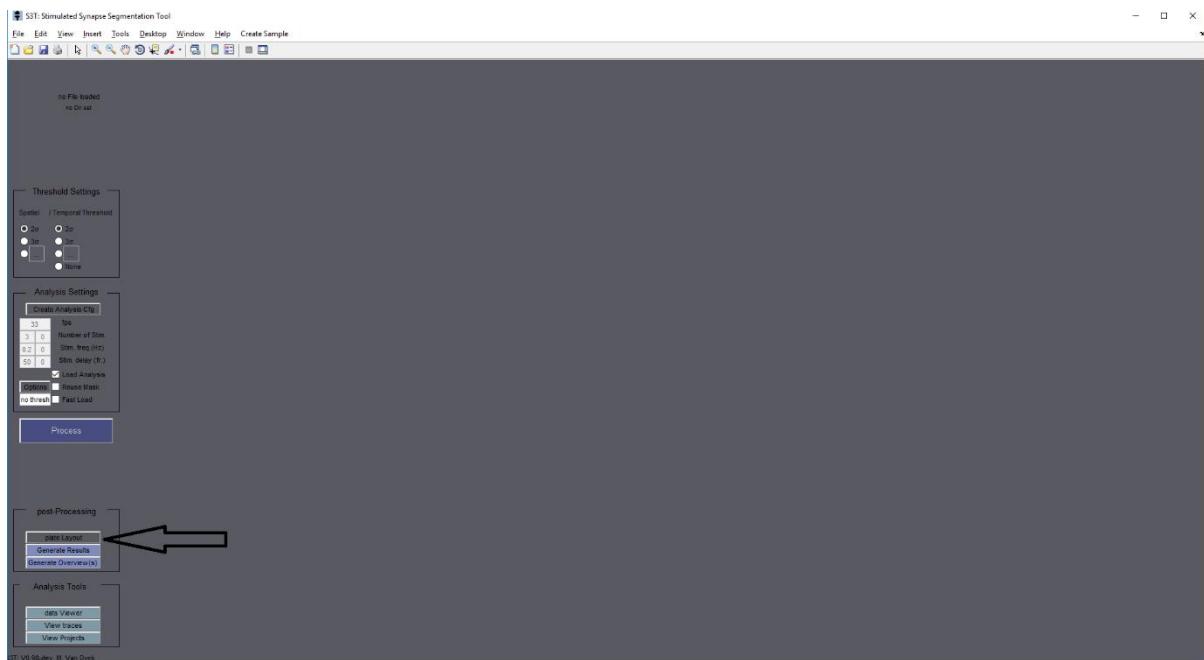
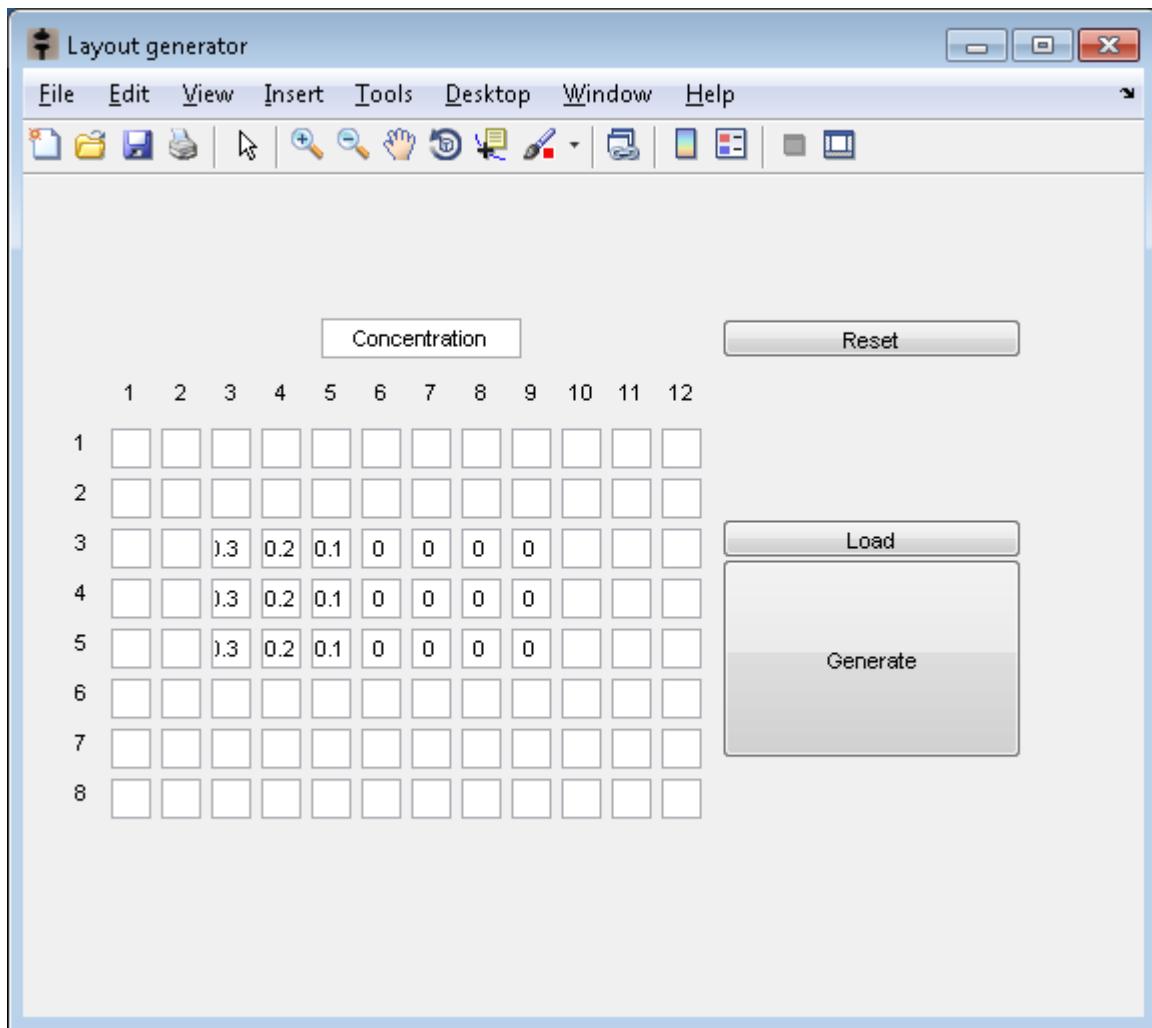


Plate Layout: This opens the GUI plate layout tool to create a plate layout when no layout file was generated by plates. This program creates csv files which represent a plate layout. For each well a numeric value can be stored. The name of the file is the identifier and should not contain spaces. Escape symbols for most commonly used symbols exist. The tool creates for the feature called concentration a platelayout_concentration.csv file which contains for all wells a concentration. Nan is used when no value is present.

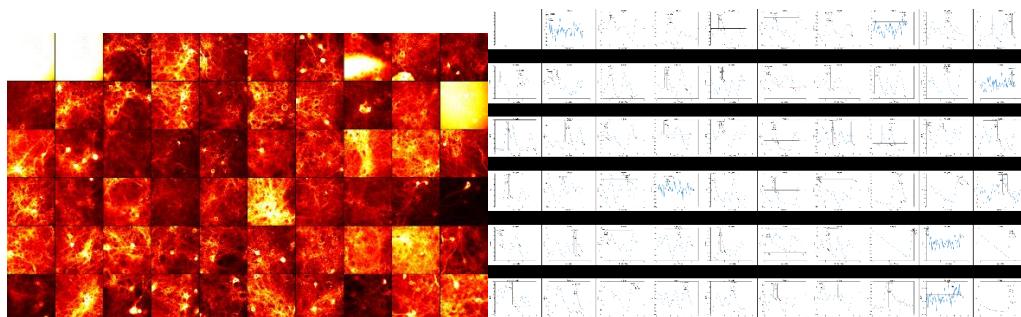


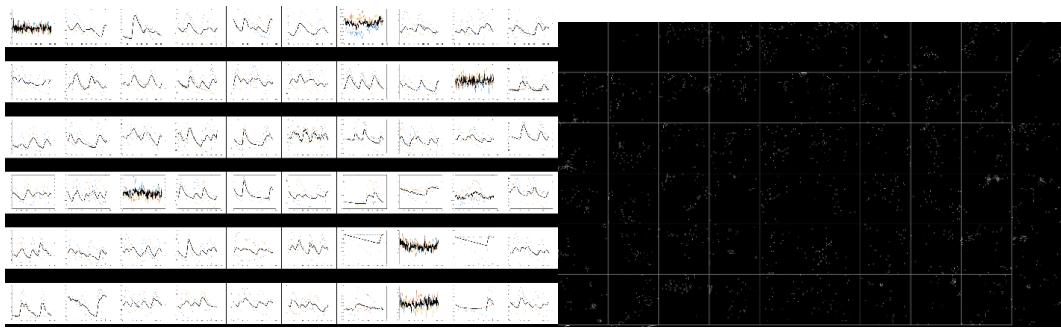
6.4.1.1.1 Read results:

Read results: This will create an AllWells.txt file containing a concatenation of the analysis of each individual well. And it will concatenate features extracted from the PLTS server. You should specify the `./output/` subdirectory, where the generated analysis files of all the wells are stored.

6.4.1.1.2 Overview generator

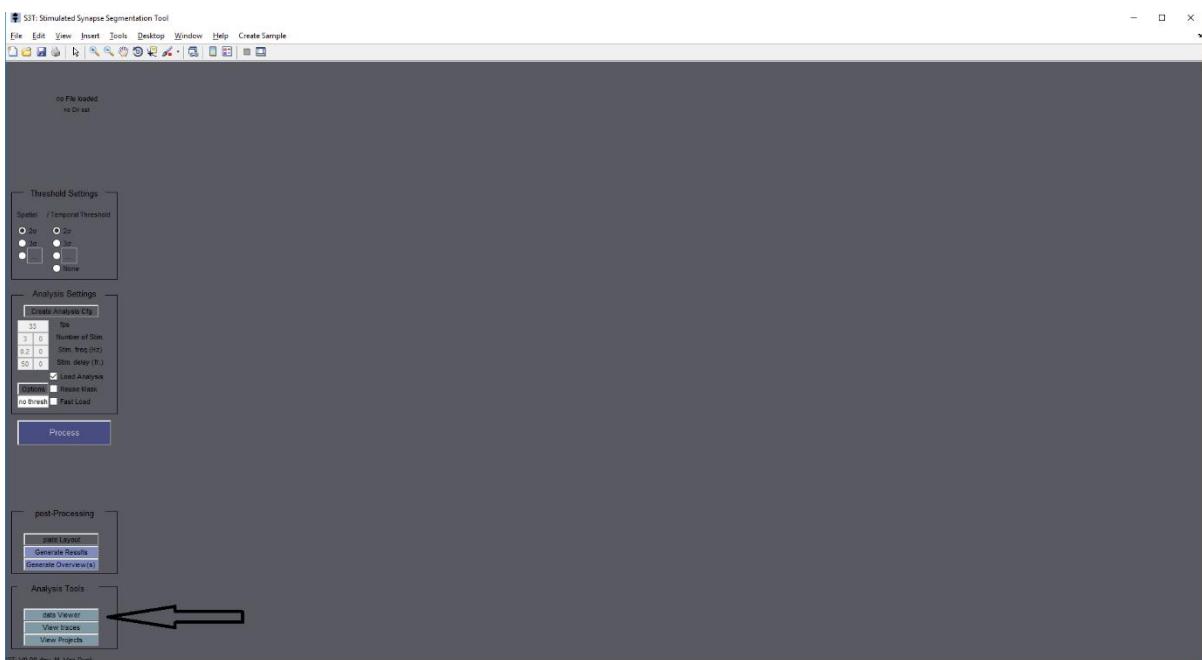
Overvie...: Overview generator: This creates a collage png picture of all the generated png files of a particular type; e.g. mask, analysis, signalPlots,..., it combines the images alphabetically sorted. (Not according to the plate layout!, To achieve this, use trace viewer)



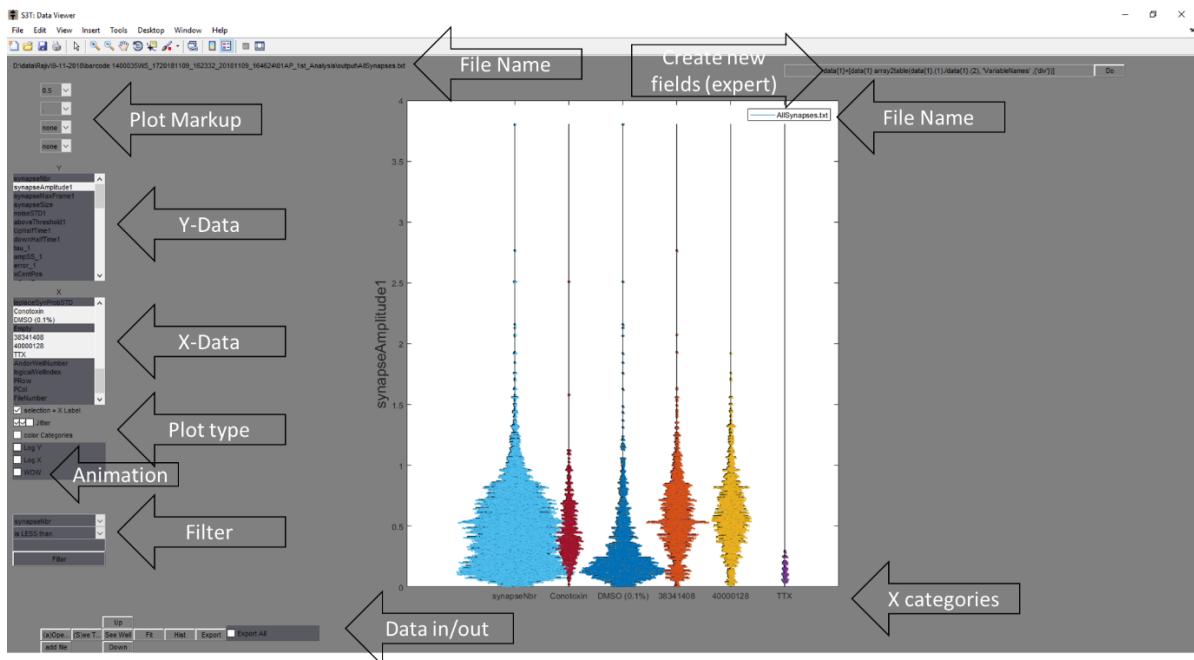


6.5 Analysis Tools

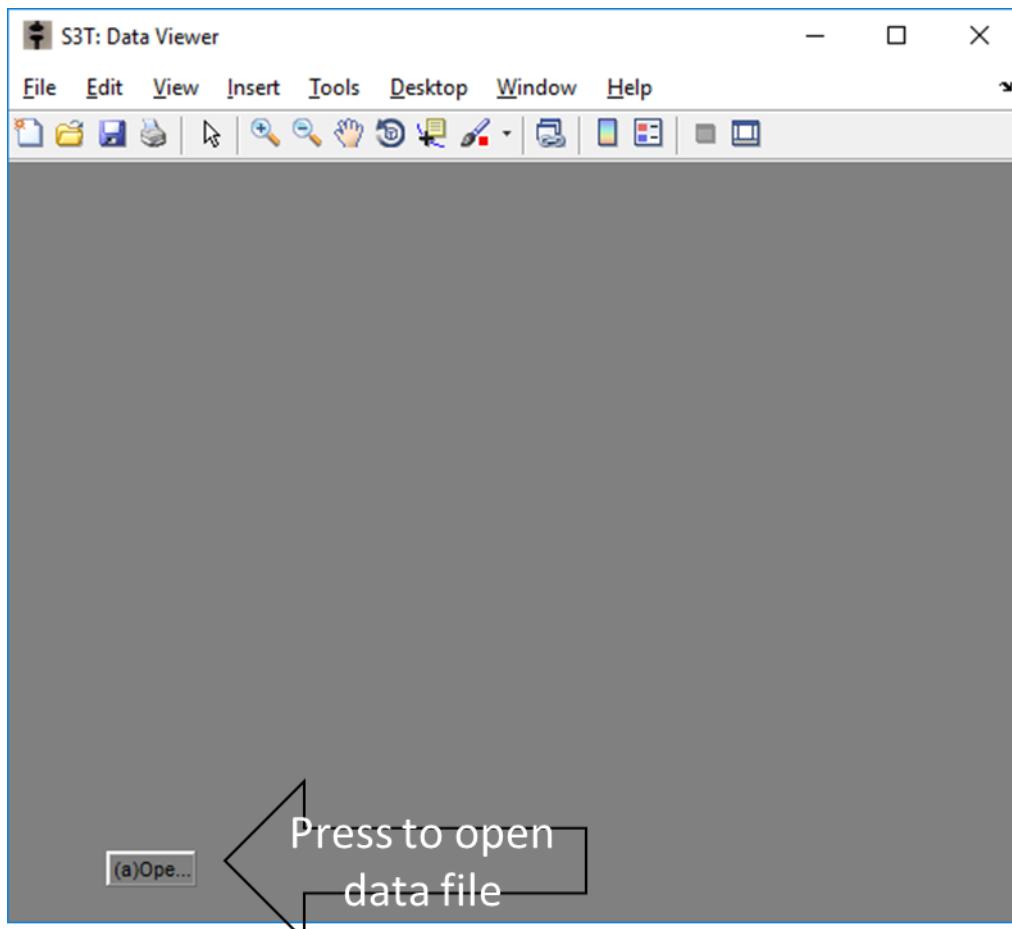
6.5.1 Data viewer

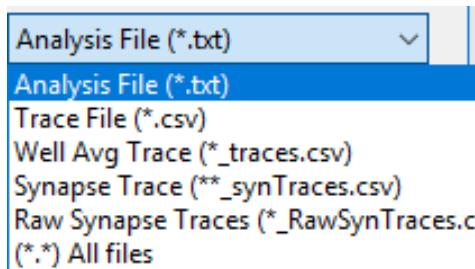


Data Viewer: Data visualization: Is a GUI data visualization tool which allows to browse quickly through the different generated result files. It allows to identify individual synapses, wells, and remove certain data-points based on different feature. Calculate histograms, export selections of the data ...



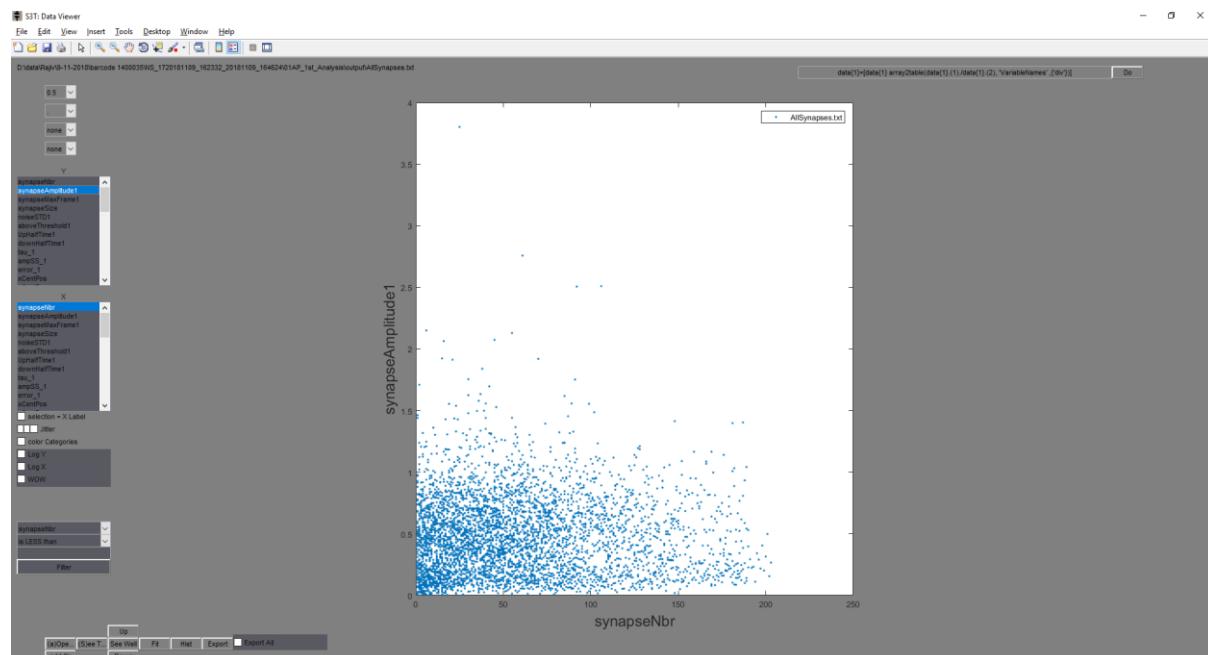
Begin screen:





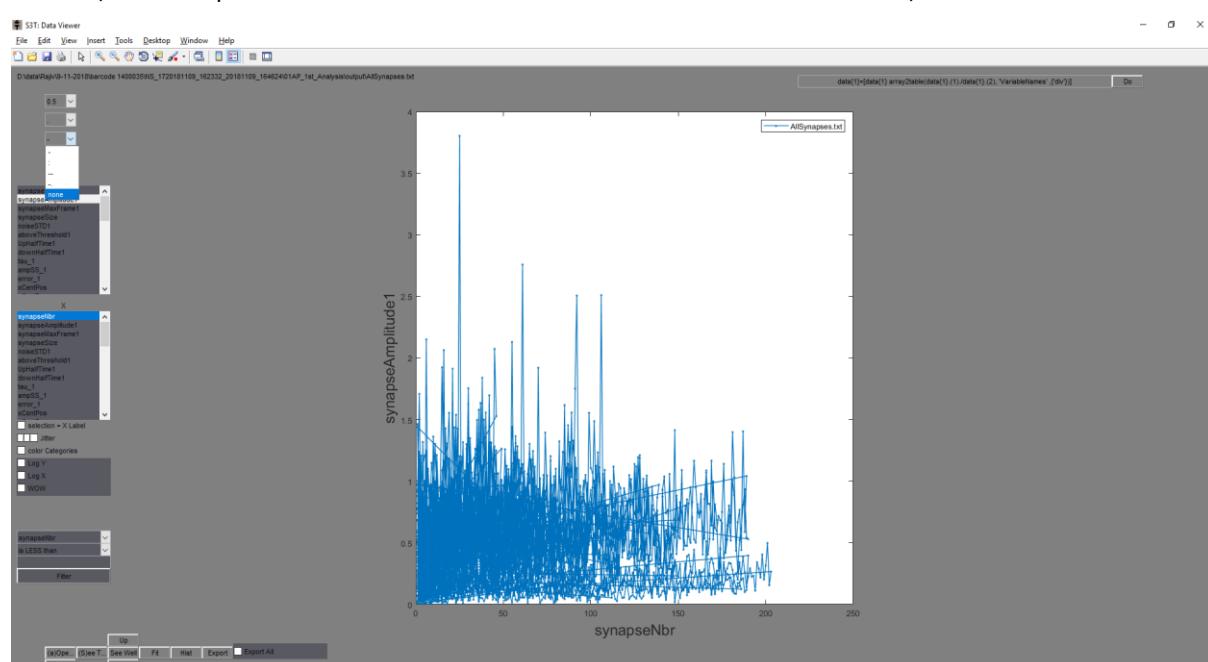
Typically, you will open a file in the output folder of the analysis you want to investigate.

Here you can choose, whether you want to open an analysis.txt file: with the extracted parameters, a trace file: or more specifically: a Well average trace, a Synapse average trace, or Raw average trace. It is possible to select multiple files. They will be shown next to each other.

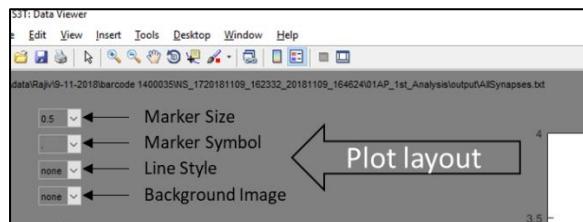


When file is opened, you can select the features to be displayed.

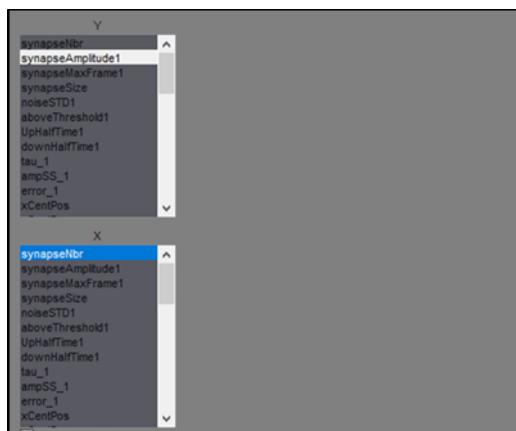
Default, the datapoints are connected: select none in the 3th selection list, bar



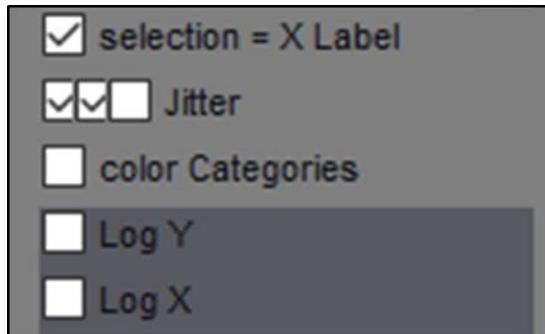
Upper 4 drop down menus allow to set point and line size and shape.



Middle 2 dropdown menus allow you to select which feature you want to have on the y-axis and on the x-axis. Multiple features can be selected with mouse+control and mose+shift button.

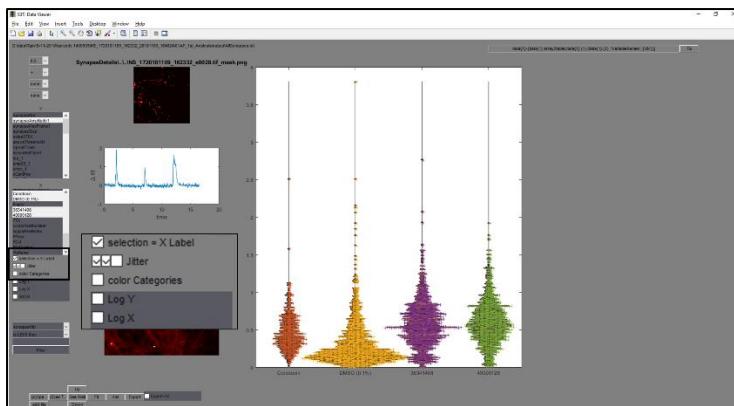


Plot Type

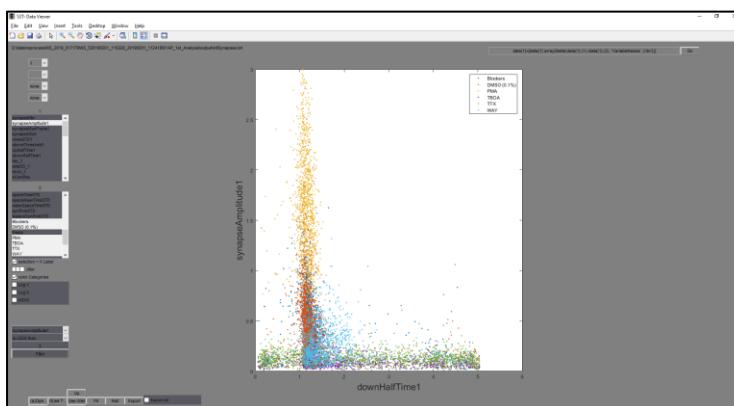


Allows to plot the different selected x-axis features as categories next to one another.

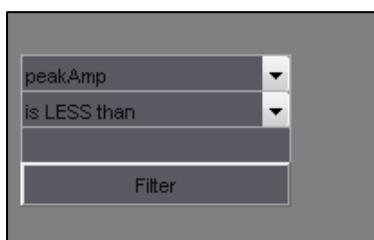
Jitter, adds (violin) plot x-noise to the data. The more check boxes, the more noise that is added.



Color categories displays the first feature on the x-axis and the other as different colored points.



Filter drop down menus allow to visualize only a certain selection of the data.



It allows hiding data-points above or below a certain value of one of the different features available for that dataset.

Open file button allows you to specify the output file you want to open and visualize. You can also select multiple files of the same type. The different files are displayed in different colors. When selecting the lowest option filename, from the x-axis features, a pseudo random generator is used to make a rain cloud plot for each imported file and visualize them next to each other.

(Remark: the plot is only updated when new x and y values are selected)

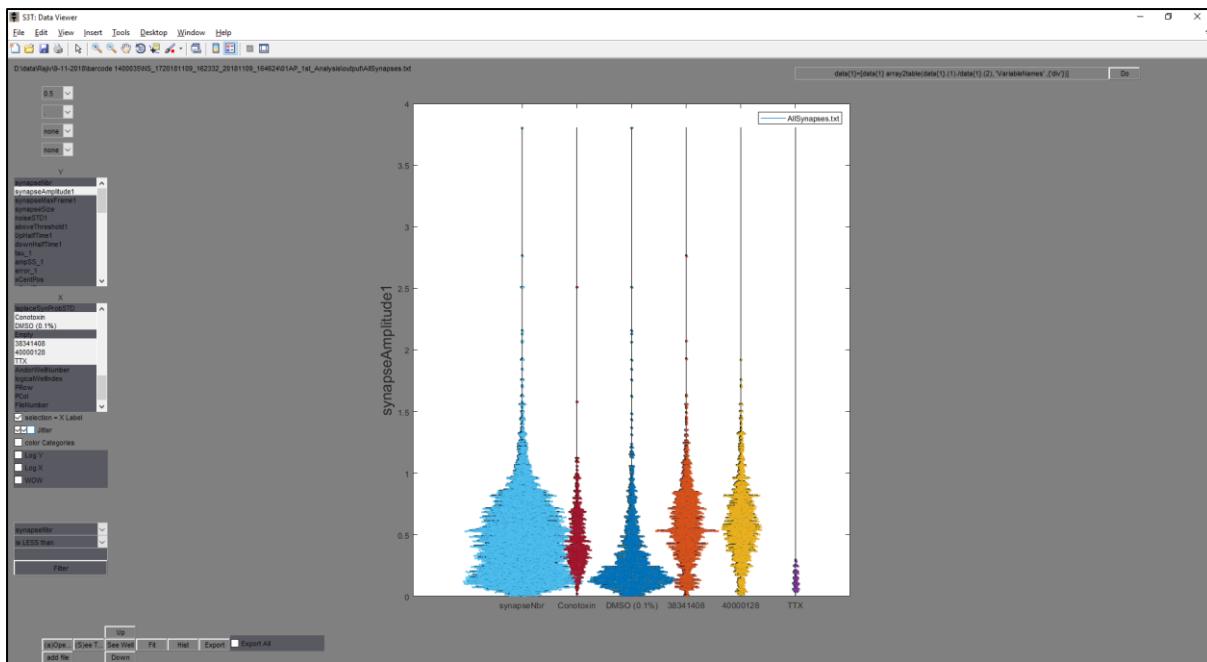
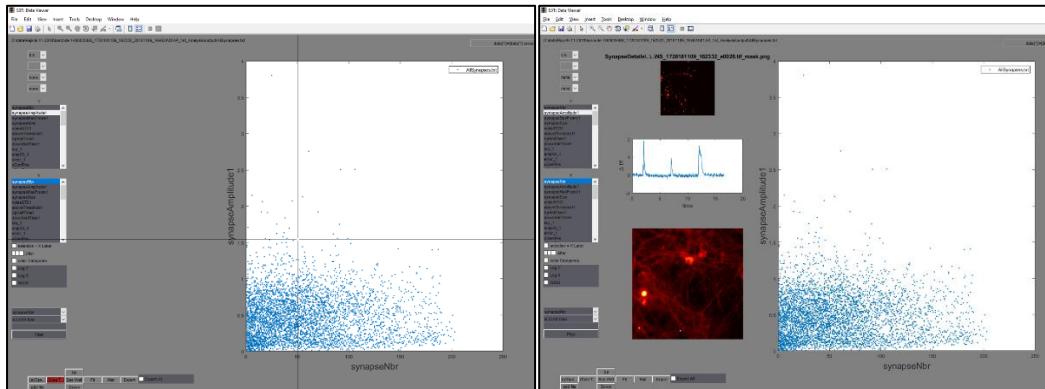


Figure...

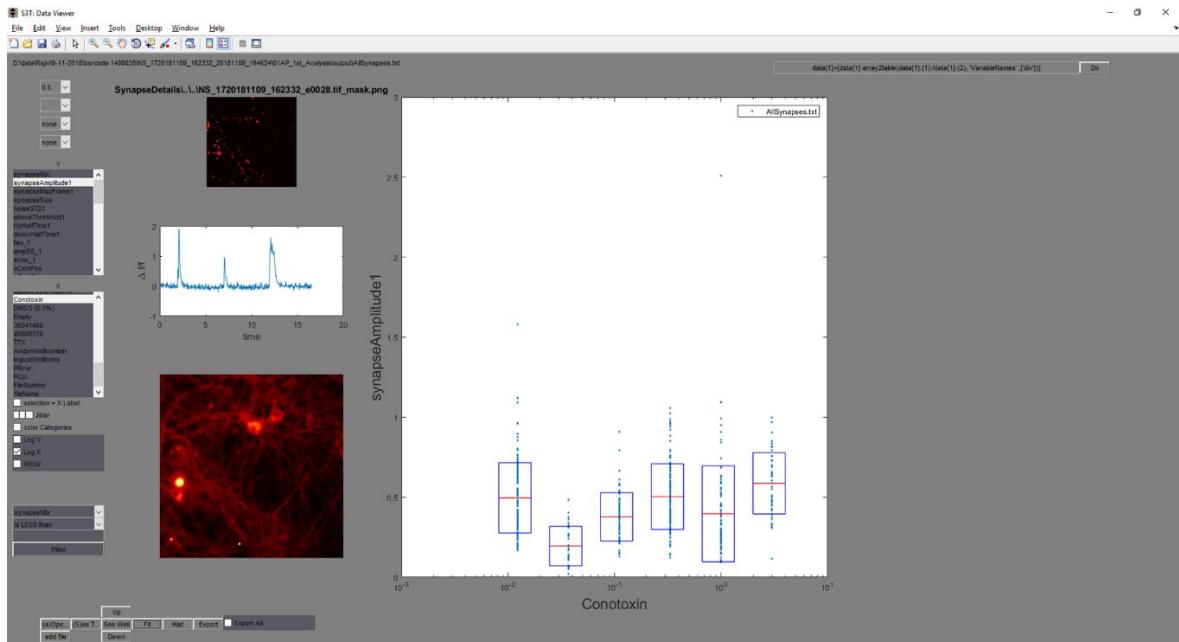
The **See trace** button (shortcut **s**) only works at certain levels in the data hierarchy. When pressed, a crosshair will show when hovering with the mouse on the plot. When selecting a well, it shows the well, time traces of all the synapses in gray and the all synapse - average in black. The mask of the well, showing the different identified synapses is shown in the top left. The time averaged well image is shown left down with LUT (Fire). When looking at an individual well, the see trace does an inverse search and allows you to select a data-point and see the location and shape of the particular synapse.



The **see well** button allows you to drop down a level in the data hierarchy to see the data of a particular well.

The **up** and **down** button allow you to go up and down in the data hierarchy. This starting from plate level down to single synapse level. (Remark: the plot is only updated when new x and y values are selected)

The **fit** button will fit a piecewise linear line with STD error bars for data points which have equal x values.



The **hist** button displays a histogram on the right side of the figure of the currently selected group. When only one group is active in a level, all data-points are blue. When data points have different colors, the histogram is calculated from the selected group.

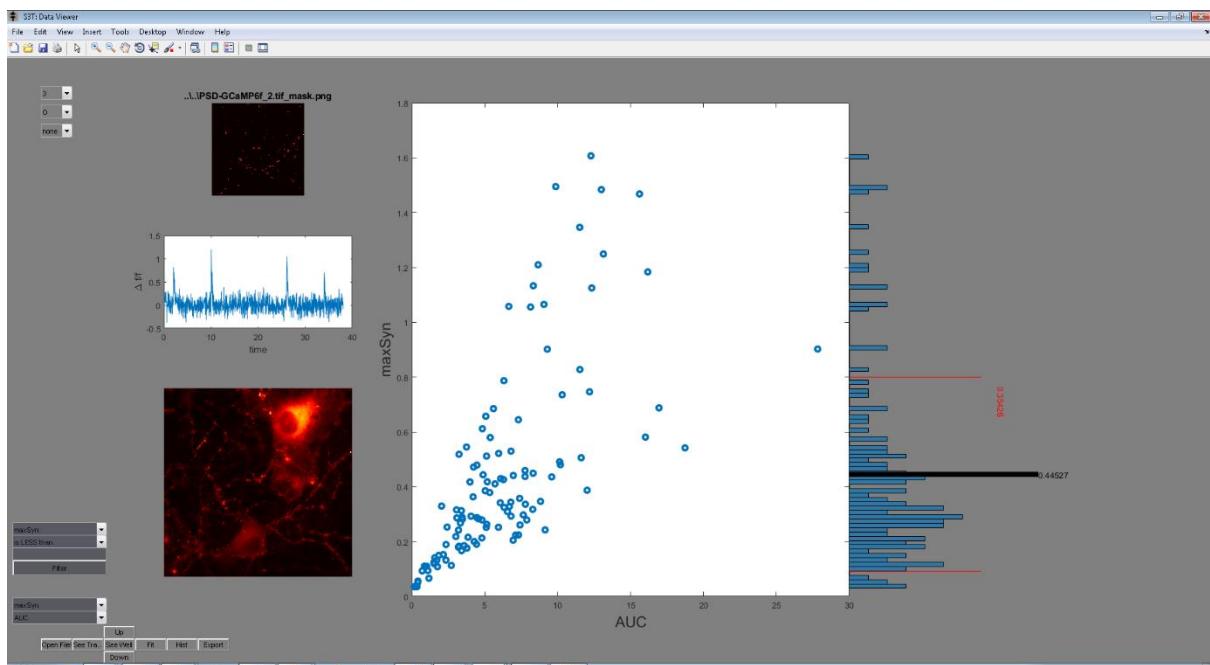


Figure...

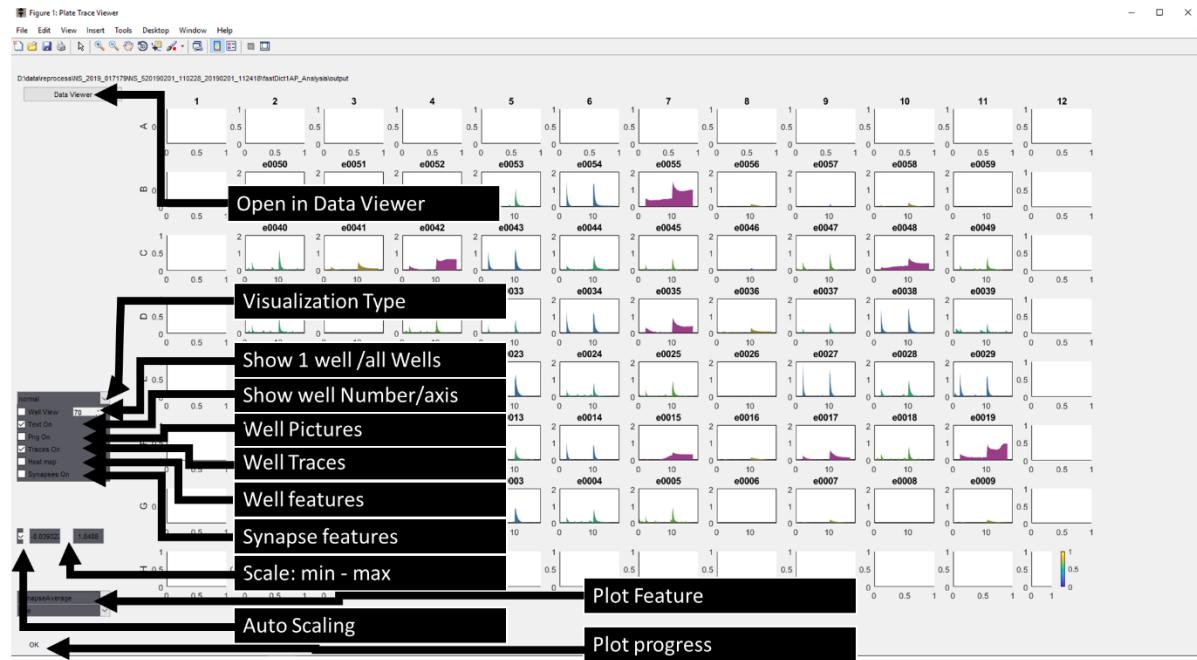
Export: The export button exports the current data points in the plot into a csv file which can be easily imported into excel, Prism Graphpad for further analysis.

6.5.2 Trace/plate Viewer:

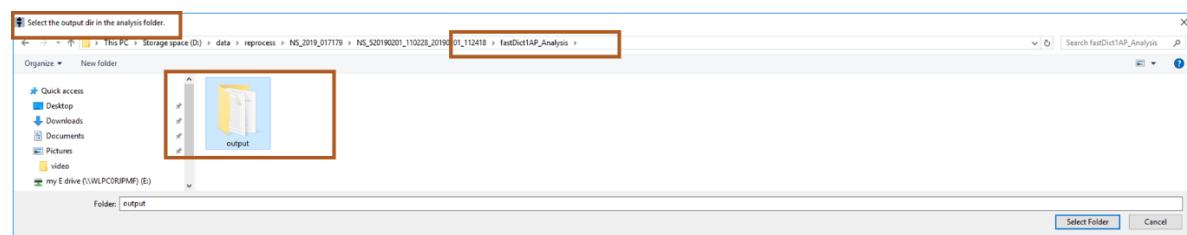
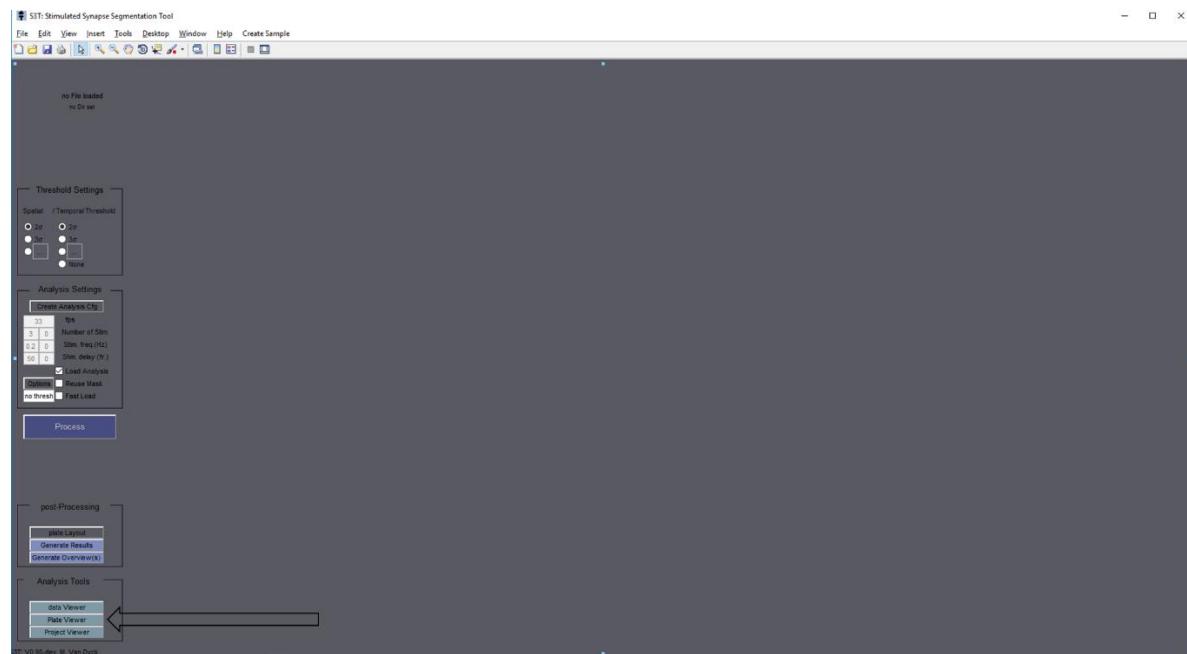
The trace viewer is an application to get an overview of the data as it was found on the plate.

When opened, you are asked to select an output folder of the analysis you want to explore the data from.

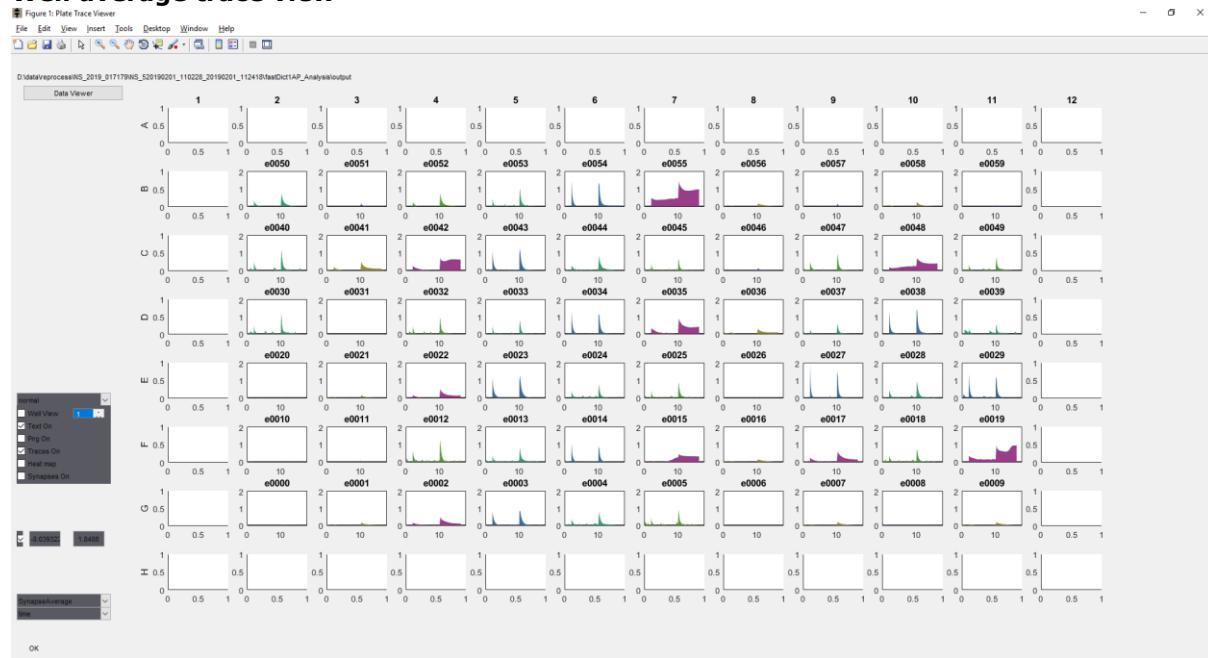
Overview features



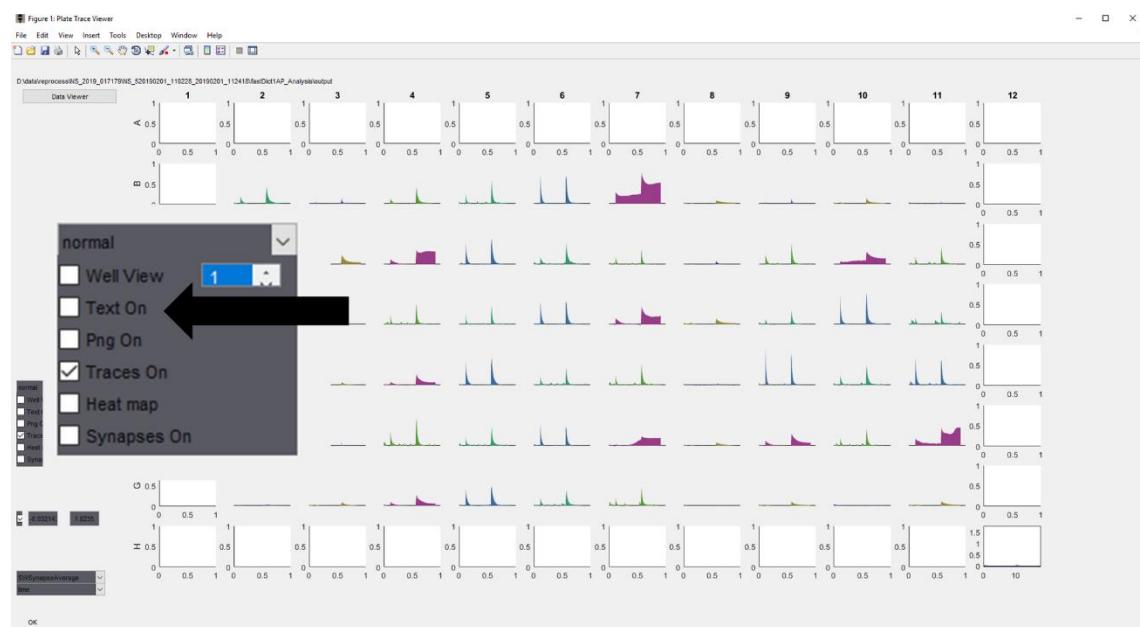
How to open



Well average trace view



Disable axis and well number

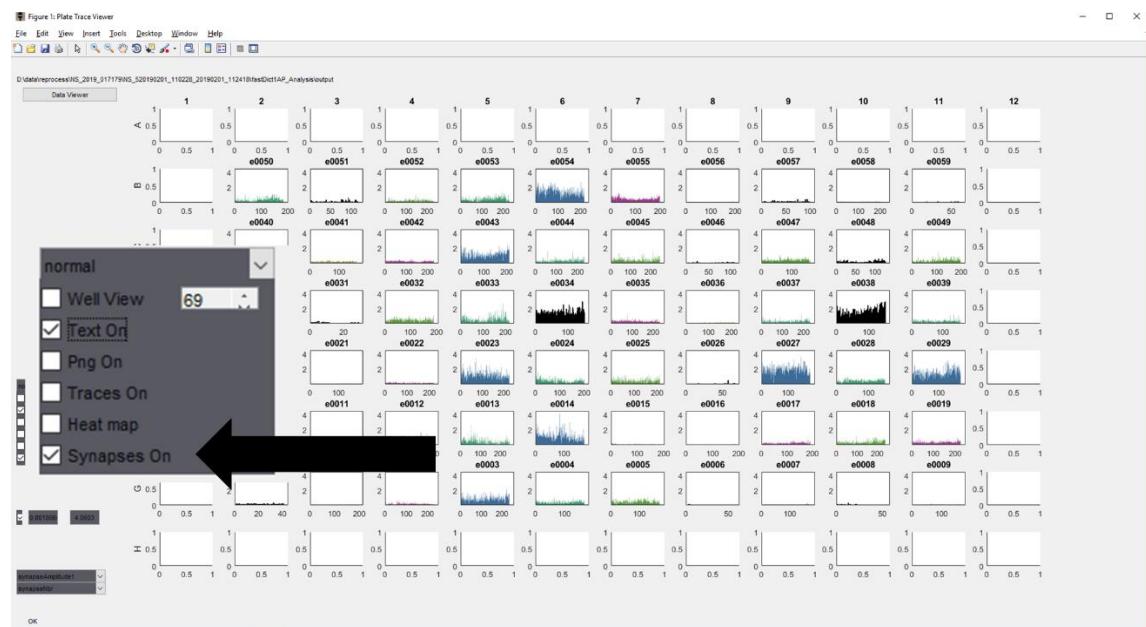


Display Well averaged features

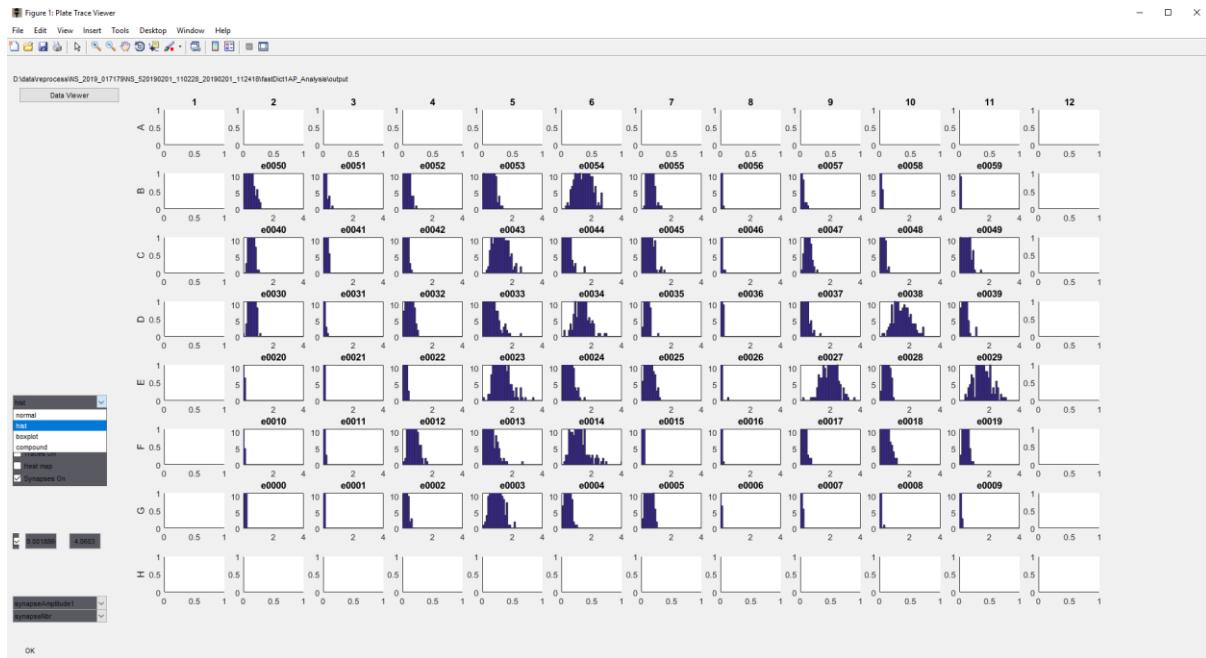


Individual Synapse data

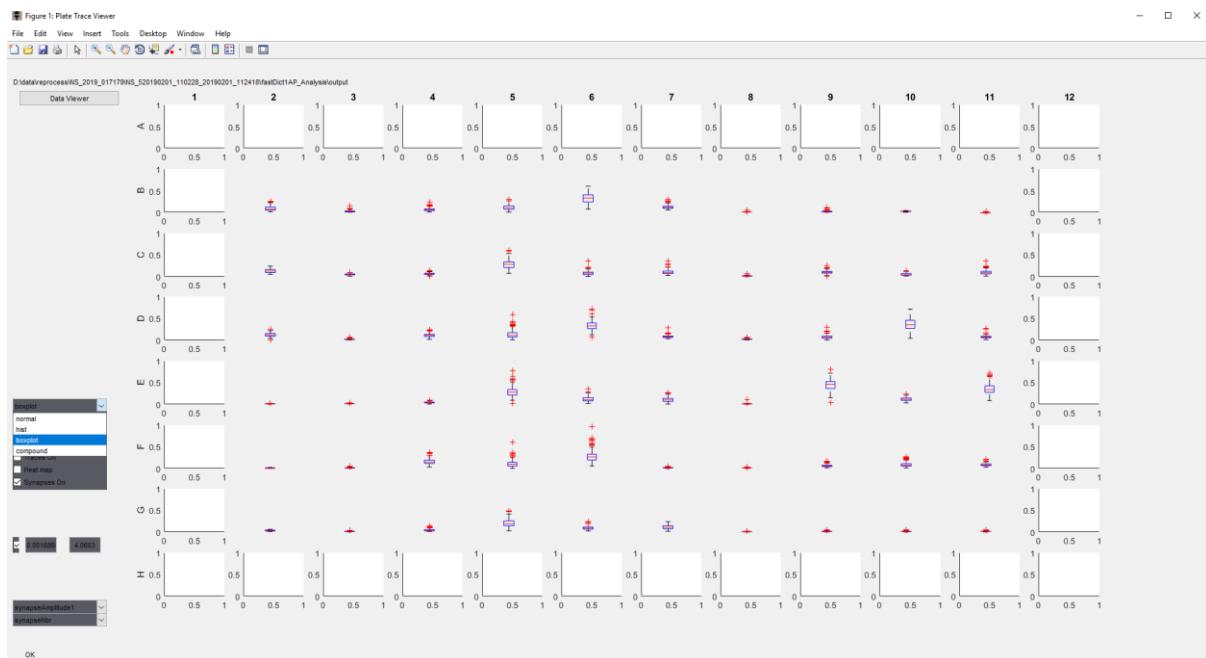
Normal



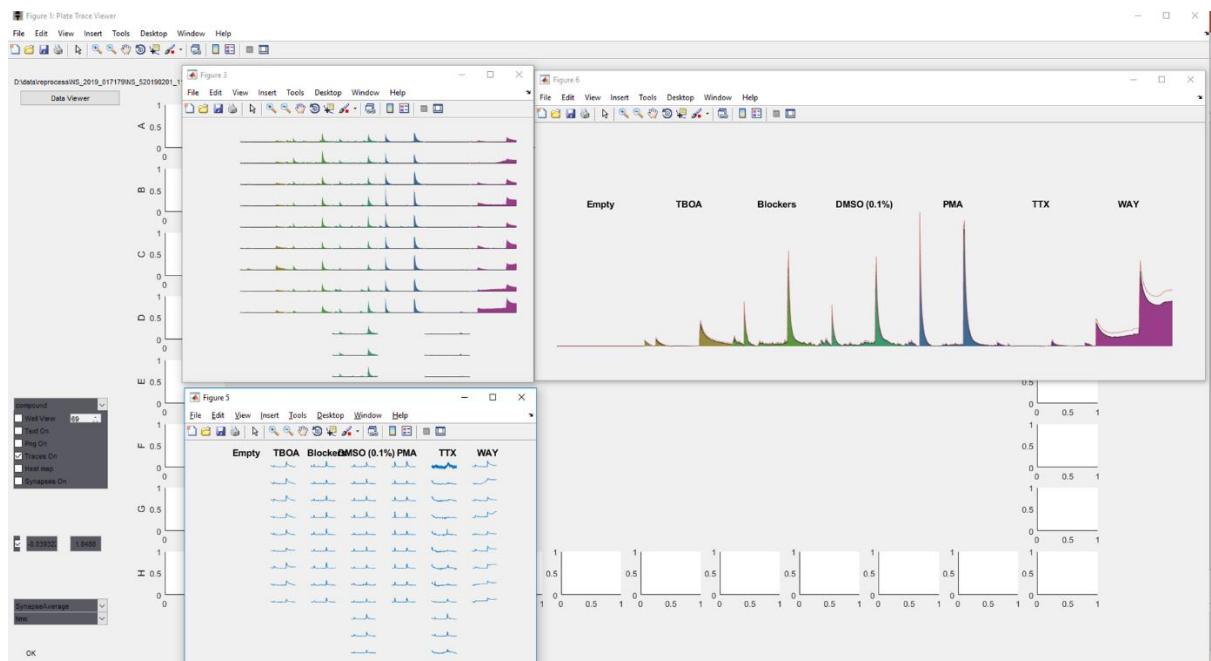
Histogram



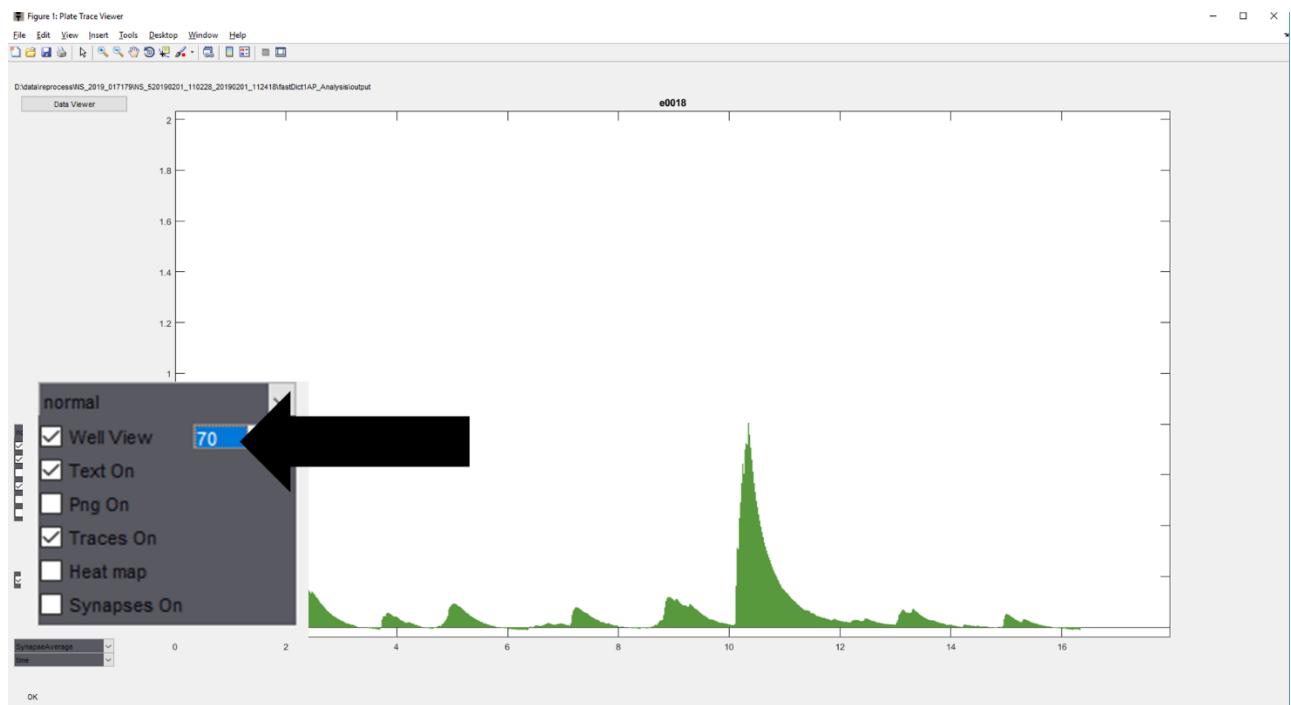
Boxplot



Compound View: Well trace example



Single Well view



Well picture view

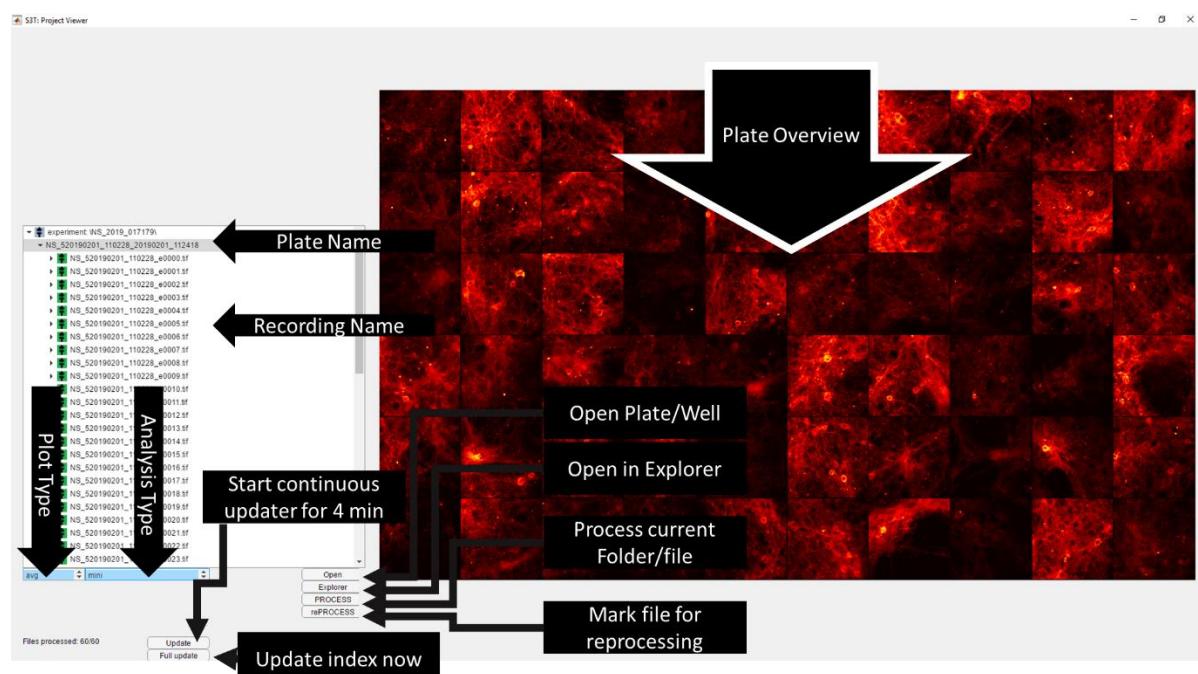


6.5.3 Project viewer:

A program to explore the TB's of data divided over multiple project folders with in each folder one or more plates with 96 well recorded movies.

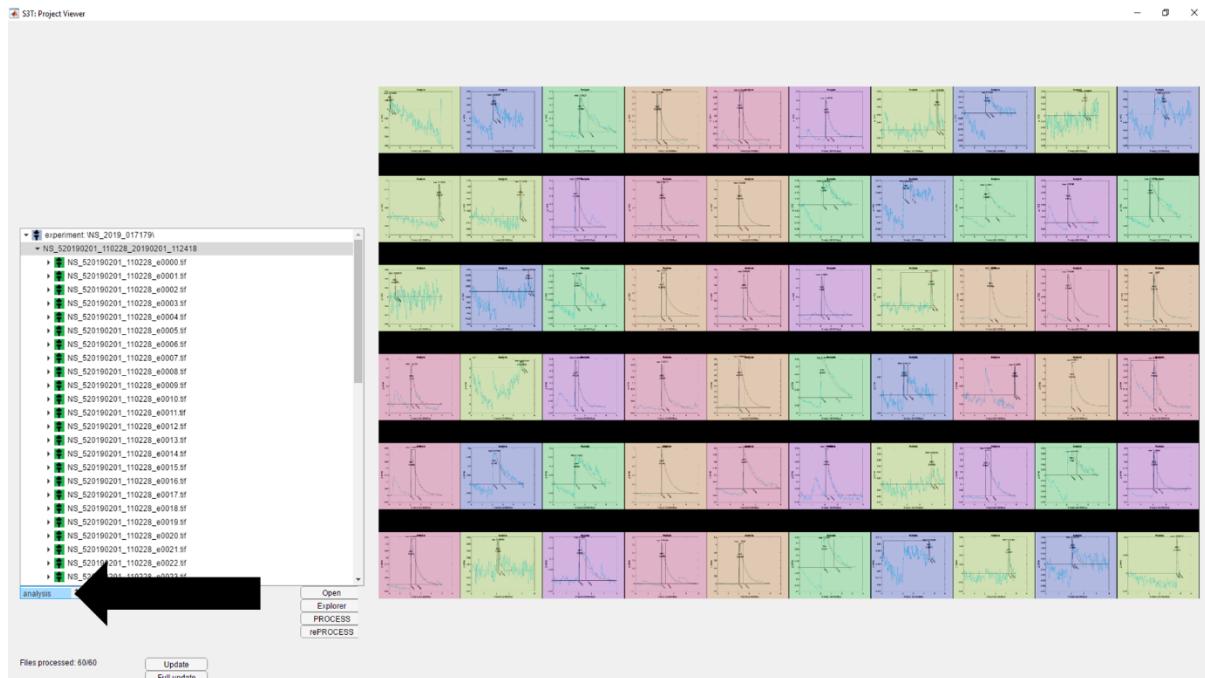
An application to index all the tiff files in your data warehouse folder and see how much of them are processed already. Scroll through the overview images generated when processing is finished, assign which movies have to reprocessed. Open plates in plate-Viewer or individual wells in Data-Viewer for more detailed analysis. Play recorded well movies, view and play plate montages. Analyze analysis overviews and plate compound distributions.

Button overview

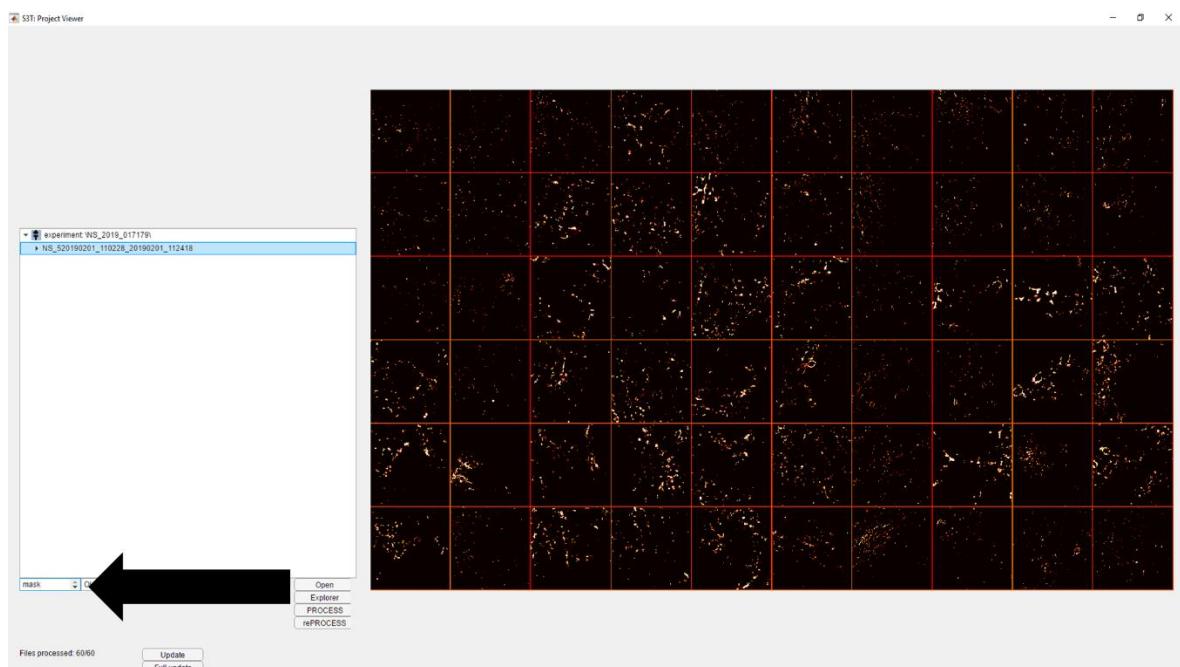


Analysis

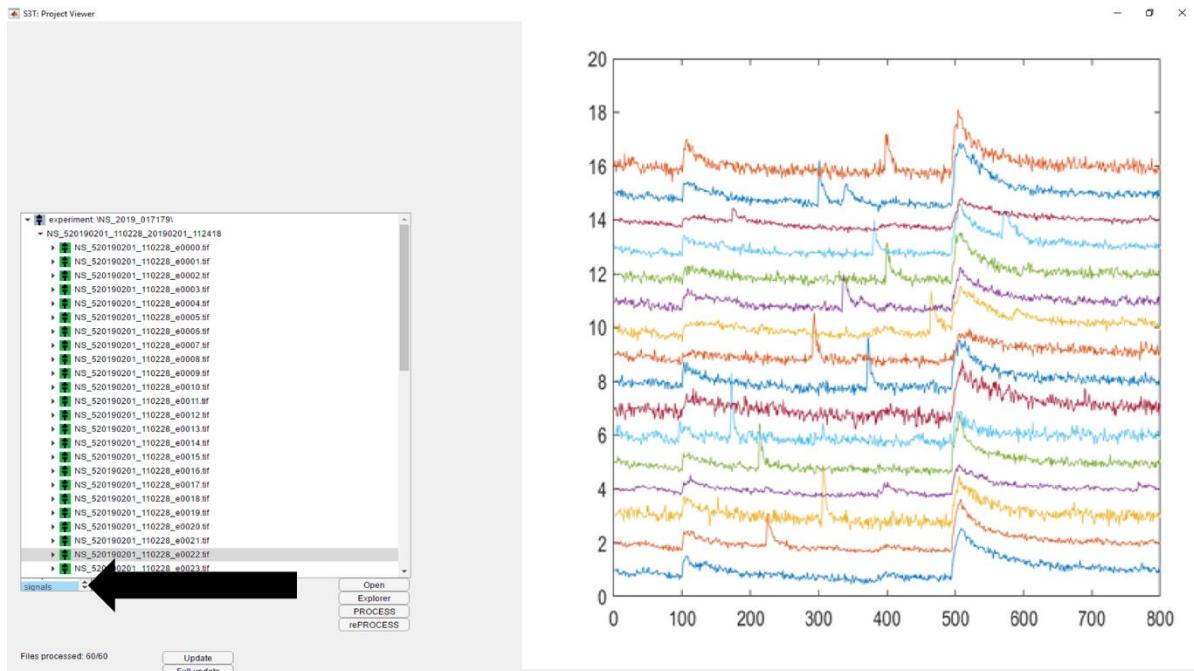
The analysis is colored by plate compound combination. If the combination is unique, the analysis is shown in a unique color.



Mask



Signals



Play

It is possible to scroll through the movies and they are played almost instantly. The movie which is played is a compressed version of the original to allow fast loadings. The movie is also accelerated to accentuate changes.

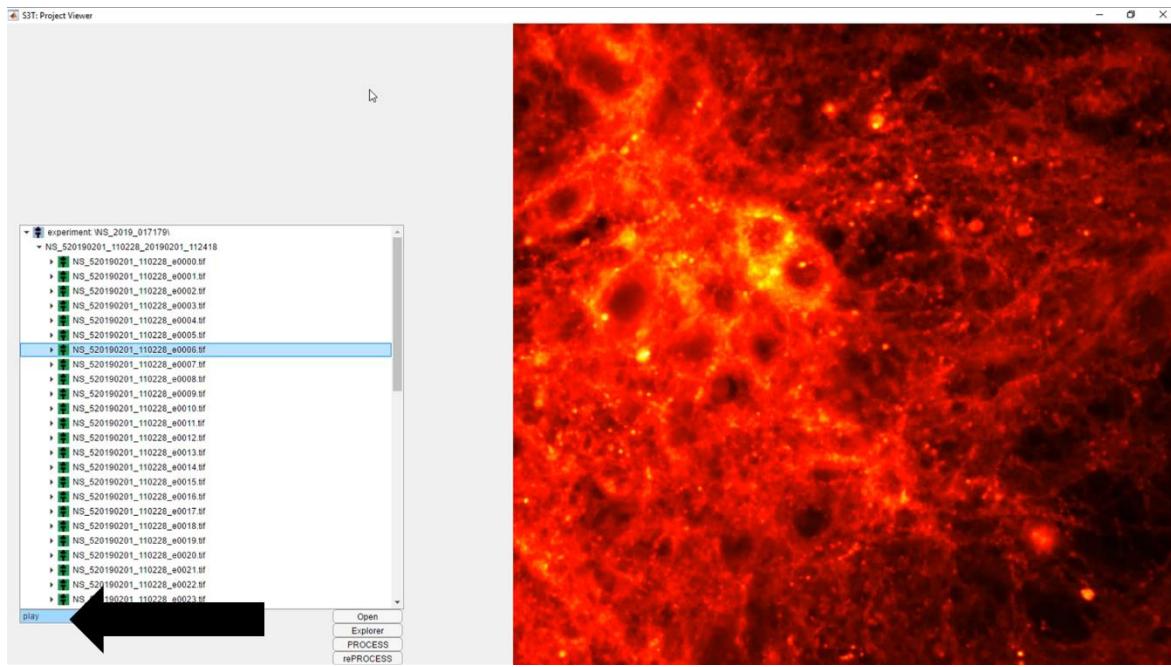


Plate Play

It is also possible to get a full plate video shown. This is also based on the compressed movies, and gives you an idea of the dynamics in the plate. The movies are concatenated from 1 to end. The plate layout is not used, so this is not the way the plate looked.

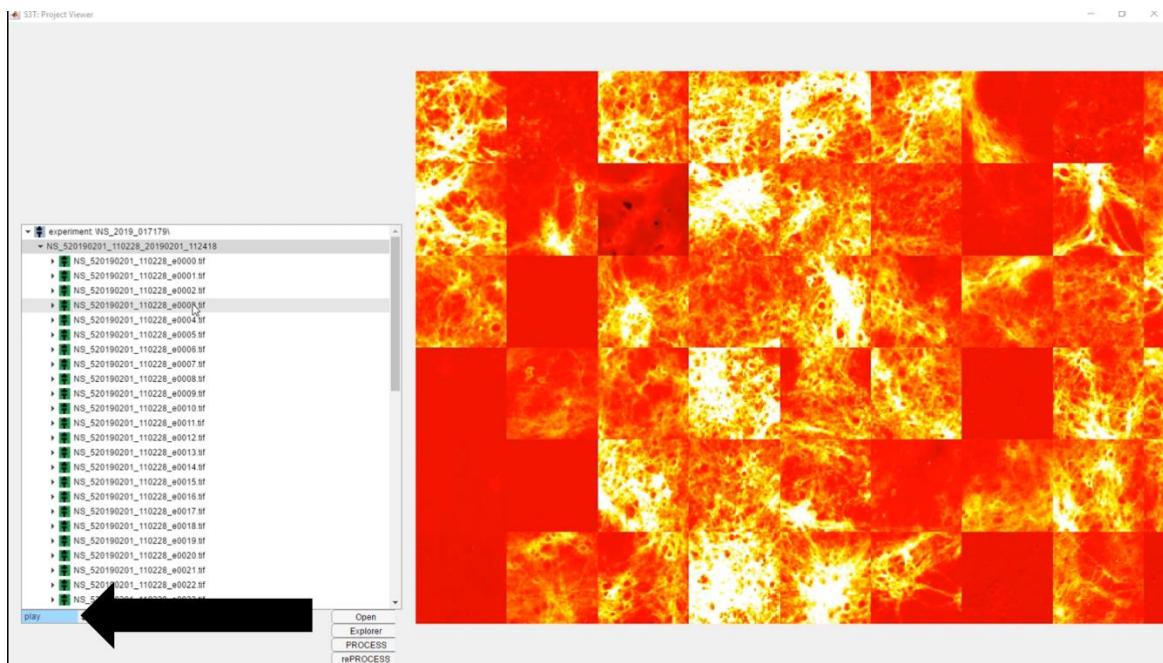
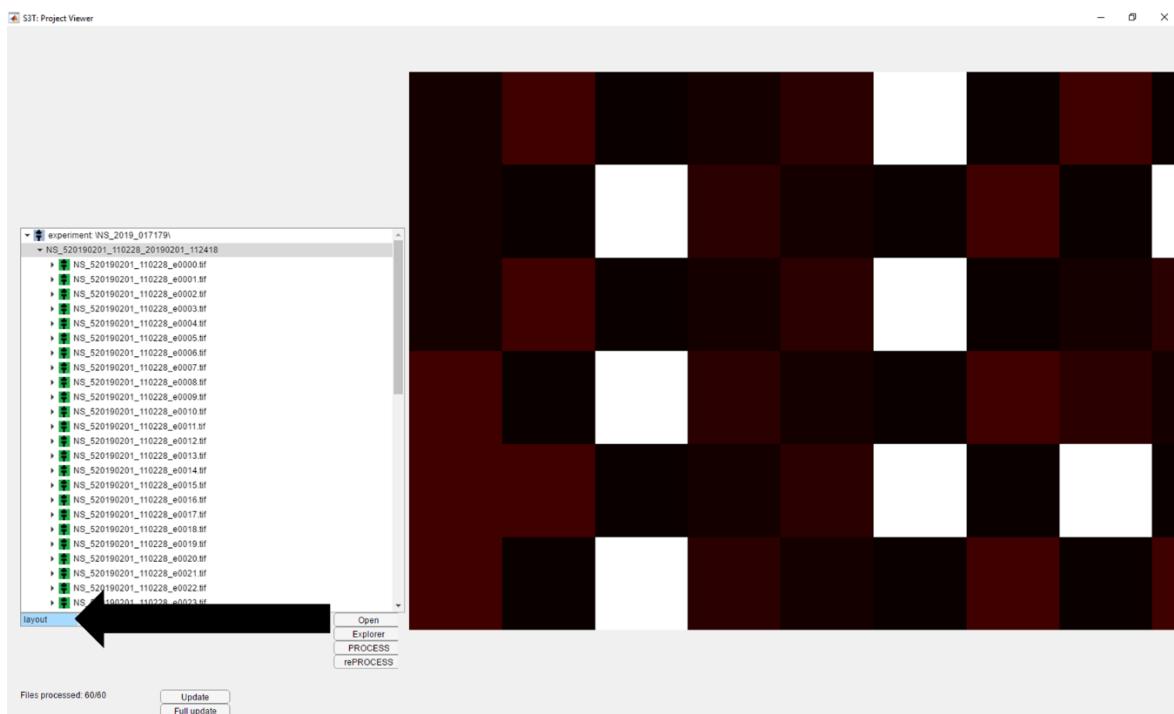


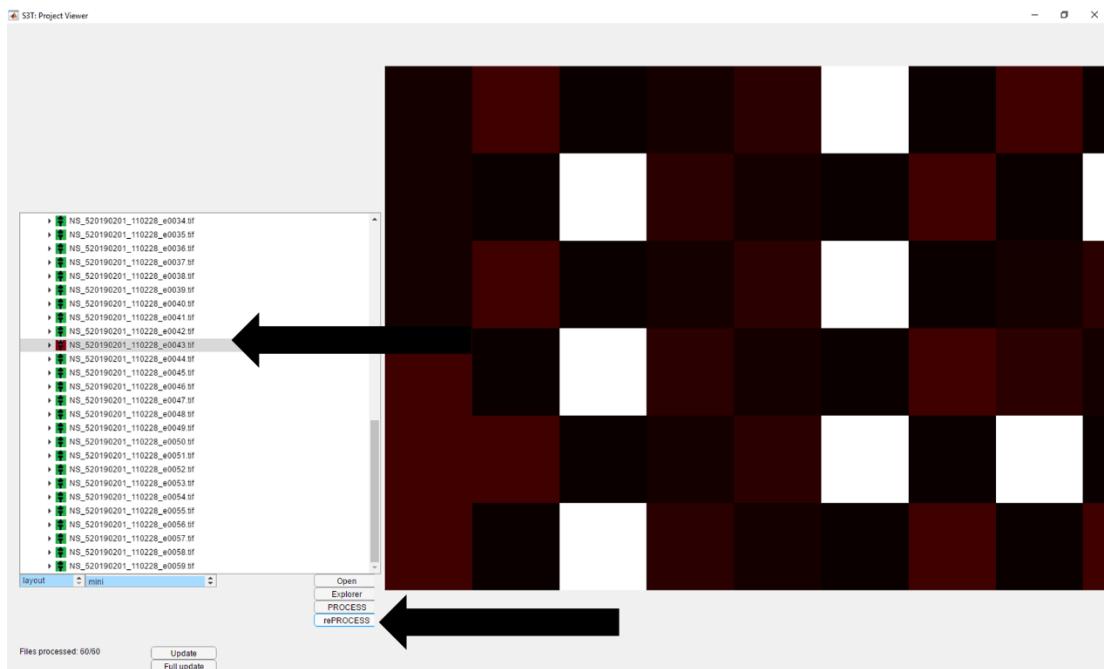
Plate Layout

A unique color is generated for each combinations of compounds which was added to the plate. Which gives you a quick overview of how compounds and compound combinations where distributed over the plate.



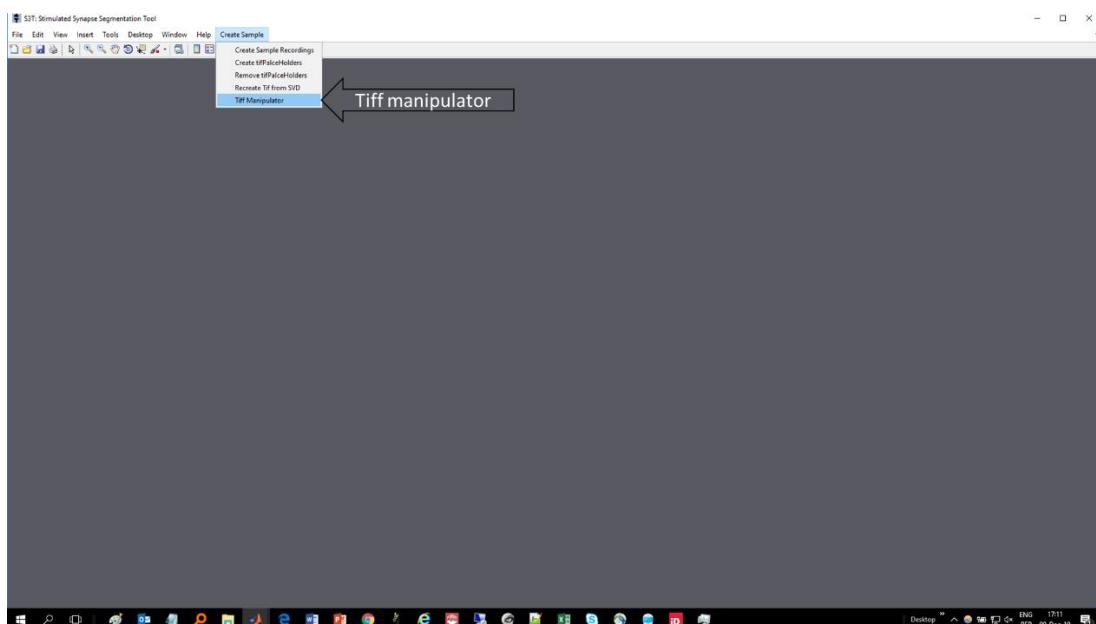
Re-process

Will mark the file to be re-processed the next time processing is done.



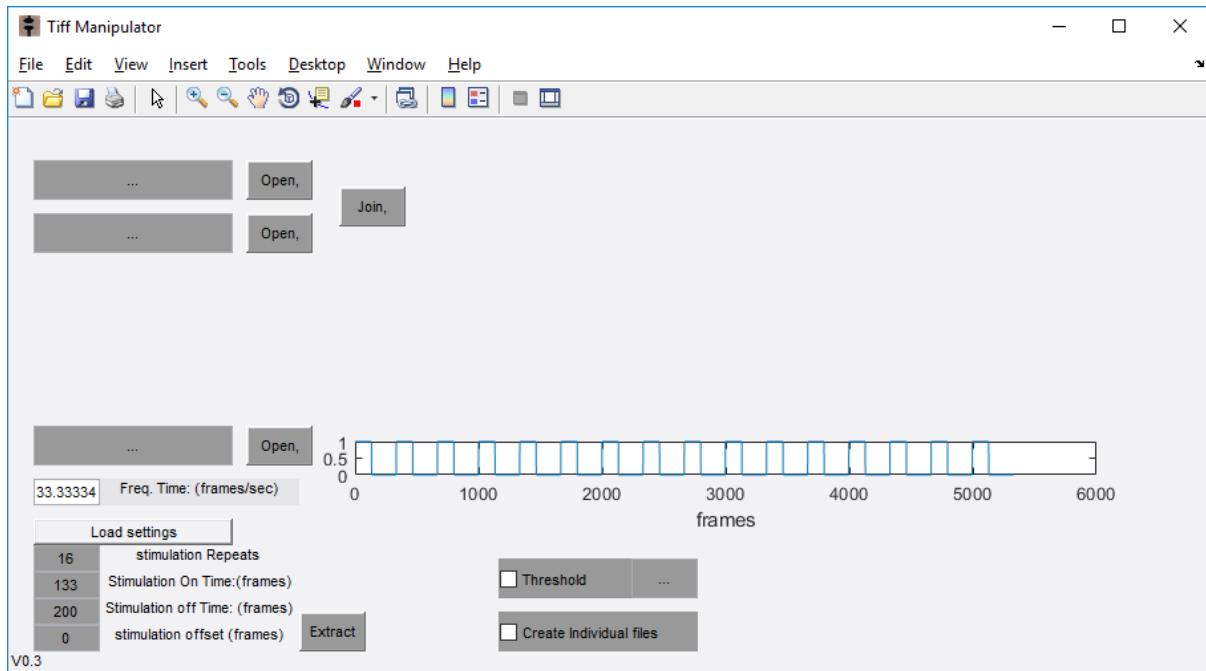
6.5.4 Database Viewer:

6.5.5 Tif Manipulator:



During recording, it sometimes happens that settings were not the way they should have been. Multiple movies should have been 1. Or 1 big movie, should have been multiple short ones. This program allows to do some elementary splitting and joining of *.tif Files.

- 1)
- 2) It allows to extract certain parts out of the movie and (Lower part)



To Join 2 movies, press the 1st open button to specify the 1st movie, press the 2nd open button to specify the 2nd movie which will be concatenated to the first when pressing the process button. When the concatenation is done the button text will change to done. A new file will be created called **comb_firstfilename.tif**

To extract certain parts out of a movie you have to define the movie you want frames to extract from by pressing the 3th open button. Then you can either specify: stimulus repeats, stimulation on time, stimulation off time, stimulation offset. Or you can load the stimulation file by pressing the **Load Settings** button to fill these fields in for you. (To result in a good extraction the stimulation file should be created with 1 channel per file, otherwise this will result into the wrong numbers in the appropriate fields).

The Freq. Time (frames/sec) field defines the frames per second which is used by acquisition of, the microscope images.

Stimulation repeats: The number of times a stimulation sequence is applied to the well.

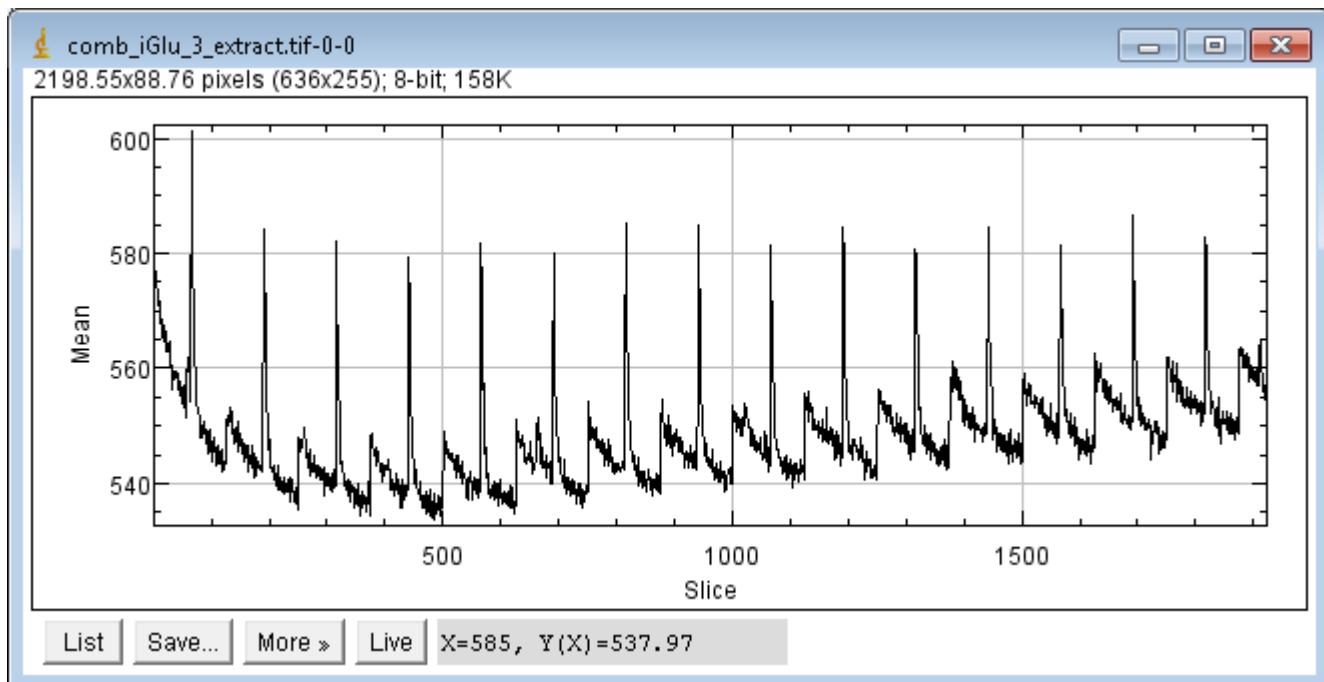
Stimulation on Time: The time a certain stimulation pattern takes. (in frames)

Stimulation time: The time between 2 stimulation patterns. (in frames)

Stimulation offset: The time to the first stimulation pattern, starting from the start of acquisition of the microscope. (in frames)

It is advised to move the movies first to your local disk since this accelerates the process a lot 45min vs 10 min. Also, we have seen certain mis-configured virus-scanners to kill the program when trying to access movies not stored on your local disk.

A typical output of the z profile looks like this.



Example:

A typical example is that we want to extract multiple temporal sequences, which are separated equal number of seconds apart. In this example we have a recording where we put on and off the microscope light multiple times, during the recording and we want to retain only the part where the light was on.

First: it not as easy as it seems from the beginning.

The main difficulty lies in the fact that the period of light on/light off is not an exact number of frames

For this there is a rounding error because you can specify on an integer number of frames. (You can't specify 0.2 frames)

So, for a particular movie we find that 15 periods of on/off sequences take 4923 frames

Dividing $4923/15=328.2$ (the recording for that particular movie and also some other was not long enough to capture the 16th light on sequence, strange)

So now we have to round off: Either the period is 328 frames with error 0.2 frames

Or 329 frames with error 0.8 frames

We chose the lowest period error: 0.2.

Now for each period we cut off 0.2 frames to early: so, $15*0.2 = 3$ frames too early in the end.

Also, in the beginning when the light goes on there is still some transient of typically 1 frame.

So, we measured the number of frames on = 129.

The number of frames off is $328.2 - 129 = 199.2$

So, in the we choose number of frames on 129, we will see in the end that we are 5 frames to early.

For this we want to reduce the number of frames on from 129 to 125.

To keep the period equal frames off should be increased: $199.2 + 4 = 203.2$ rounded to 203.

Finally, most importantly

When we make the period, we cut too early in, so we should start 5 points later in the beginning to combat this.

So, for this the numbers to put in the tiff manipulator are: On=125, off=203, offset=5

In short:

4923/15=328.2 PERIOD
1443-1314=129 ON
328.2-129=199.2 OFF
15x0.2= 3 frames rounding error shift
129-4=125
199+4=203
shift 5

6.5.6 plate Meta Data Converter

This routine searches for a barcode.txt file inside the data folder.

The first item in the barcode.txt file should be the barcode number which is submitted to the PLTS server.

The server location should be configured in the urls.json file. (for more information see appendix: urls.json)

A JSON file is returned, which is than parsed and the sample Code and concentration are used to generate Platelayout_SAMPLECODE.csv files for each of the sample codes found in the plate.

6.5.7 Analysis configuration file(s)

The type of analysis done on each movie is defined in the analysis configuration files.

These files are stored next to the image data files. The files should be named ..._Analysis.xml

Multiple Analysis.xml files can be placed in 1 directory. All analyses are run on each movie.

An Analysis file can be created with the Analysis configuration Tool. An Analysis file can also be duplicated from a previous project and modified, since these are human readable xml files.

If a certain movie needs a special analysis, then an exception can be made for a particular movie by creating an Analysis file named preceding the tif file name. E.g. for *NS_1720181109_162332_e0008.tif*, a special analysis can be defined by creating a *NS_1720181109_162332_e0008.tif_ANYNAME_Analysis.xml* file. The other analysis files in the directory will be omitted for this *.tif movie. The output of the analysis will be stored in the *NS_1720181109_162332_e0008.tif_ANYNAME* folder.

6.5.8 Analysis configuration Tool

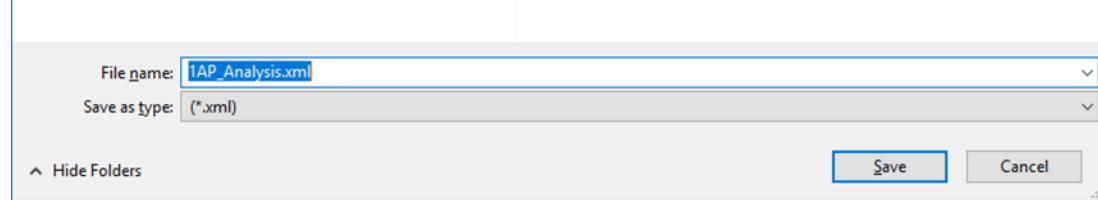
Opened with the **create Analysis CFG:** button Is an analysis configuration tool to generate xml files containing the field numbers needed for the analysis. E.g. fps, number of stimuli, ...

When generate is clicked, a popup window opens to specify where to **store the analysis file.** All **tiff data in the directory** is analyzed according to that analysis file. If multiple analysis files are stored in the directory, all files will be processed with **all analysis files.**

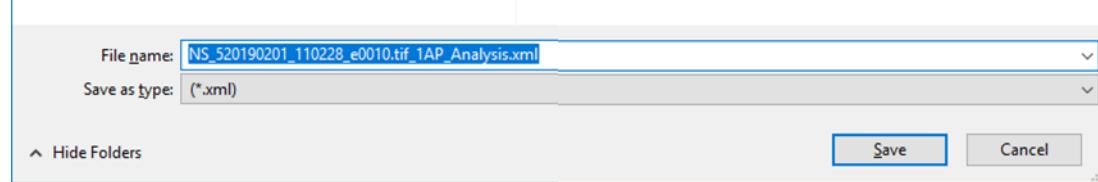
When the analysis file contains .tif in the name, the analysis is **only** run on **that particular *.tif** file. All other analysis files are run on all other tiff files in the directory.

If a tif file has a **specific analysis file** for it, the directory wide analyses is **not run for that file**.

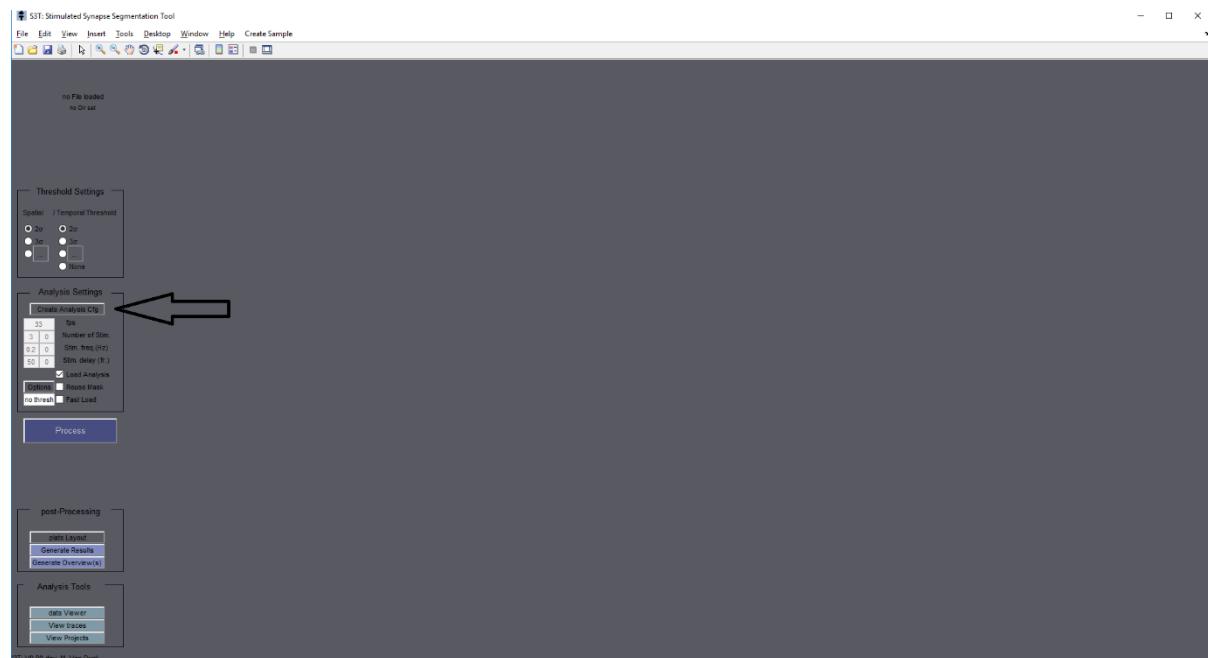
Analysis file for all *.tif files in directory

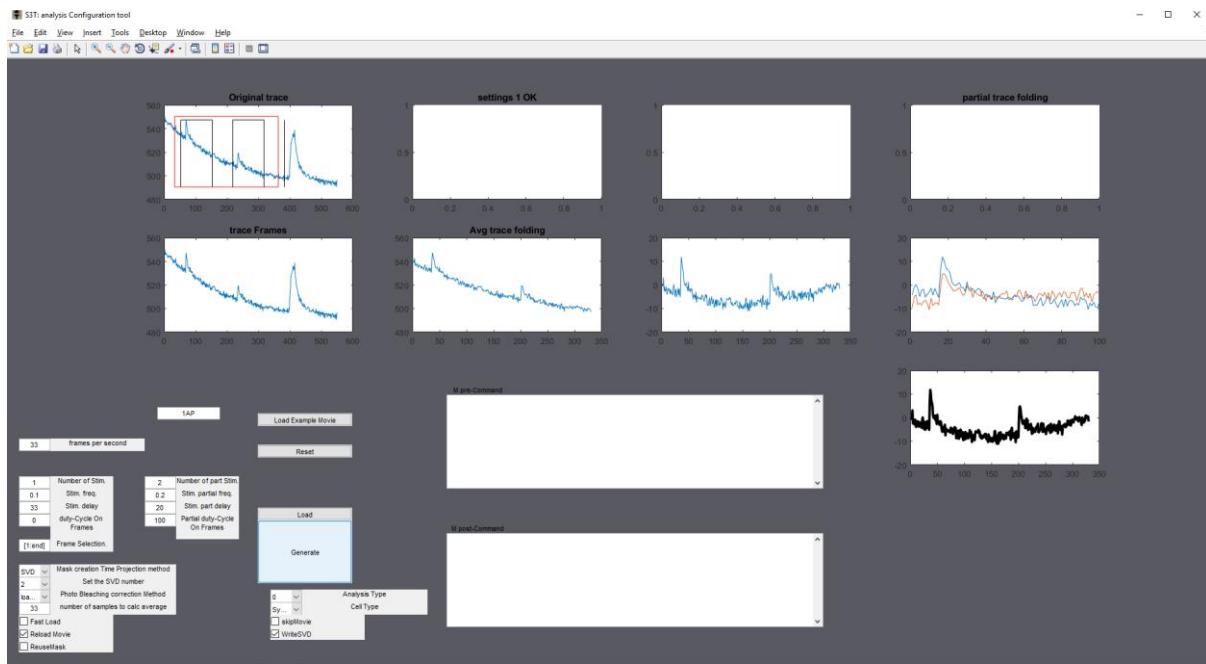


Analysis file for 1 specific *.tif file (filename pre-pendend)



For more advanced users, more analysis options are available by editing the analysis files in a text editor. For details see Analysis configuration file syntax.

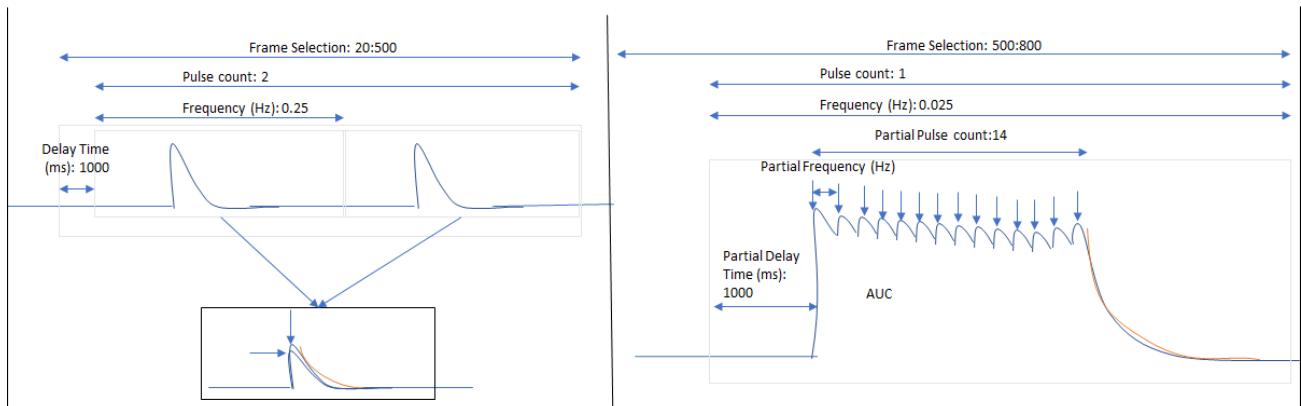




Analysis configuration file syntax

Different parameters can be defined to configure the analysis done for each movie.

- Load Example Movie: When you load example movie, the settings are visualized on the average trace of the movie . As visualize above. In the first window you see the full trace. In red the first analysis window. In Black the second sub analysis window. The difference is that we assume that the beginning and the end of a red window, (first column parameters) the trace is at base level. When we select the black window (second column parameters), we just look for the local response in the window, but don't estimate the fluorescence baseline (F_0) used in the $(f-F_0)/F_0$ calculation.



- Camera Exposure Time (s) - 0.03000

The frame rate with which the frames were captured.

- Frequency (Hz): 0.2500000000000000

The frequency with which the temporal pattern is repeated. 0.25=every 4seconds

- Delay Time (ms): 6000.002070

Delay from the beginning of the movie to the start of the temporal window of interest.

- Pulse count: 1

Number of temporal repeats to be averaged out</info>

- Partial Frequency (Hz): 0.2500000000000000

The stimulation frequency during burst

- Partial Delay Time (ms): 0.0

Delay from start window to start burst stimulation

- Partial Pulse count: 1

The number of spikes in the spike (burst) train

- Eigenvalue Number: 0

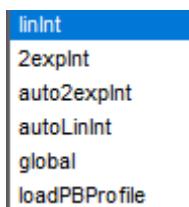
Selection from the eigenvector to create mask from, choices: 1 or 2 or 3 or 4, For interactive selection during processing set 0: A popup window will appear for a few seconds to make your choice.

- Reuse Mask: 1

choice: 0 or 1, create new mask = 0 or reuse existing mask = 1

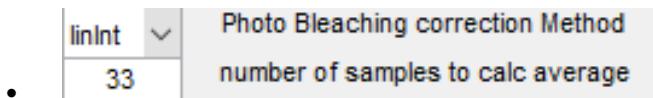
- Photo Bleaching:

choices:



- linInt: Linear interpolation between first and last points of window of interest.
- 2expInt: fit of sum of 2 exponentials + offset, between the first and last 33 points of the window of interest.(Number of points can be changed with number below)
- Auto2expInt: Fits sum of 2 exponentials and tries to automatically detect the points to use to fit in the Window Of Interest.
- Autolinint. Linear interpolation and tries to automatically detect the points to use to fit in the Window Of Interest.
- Global: fits a 2exponential to the average trace of the FOV, the uses a ratio metric method to photobleach the entire movie.
- LoadPBProfile: A photo bleach profile can be loaded to globally correct for photo-bleach, The profile is stored at the installation directory in PBprofile.dlm.

The number of samples used in the begin and end of the window to fit linear or 2exponentials can be adjusted with the second parameter. When used auto methods, this parameter is discarded.



- Frame Selection:(1:end)

Selection of the frames of the movie you want to use for mask creation. The temporal window must be inside the frame selection, otherwise a new analysis with Reload Movie: 1 should be done.

- Skip Movie: 0

Defines if you want to skip the movie for processing or not.

- Write SVD: 1

Defines if you want to write the SVD to the ./eig/ subdirectory or not.

- Reload Movie: 1

Defines if you want to reload the movie after the previous analysis.

When using frame Selection, different from the default (1:end), frames from the original movie are removed. If the next analysis needs these frames, a reload is required. If no reload is done, the frames, delays, ... should be defined in function of the frame stack resulting from the previous analysis.

- Analysis Type: Default 0
- The idea is that you can create a library of named configurations settings and can load them here.
 - Default 0: Will use the settings
 - Stimulated: not applicable yet
 - Spontanious: not applicable yet
 - Minis: (Experimental), will discard most other settings, and will look for mPSCs in the time frame: 150:5:470;

6.6 Files

This section describes the purpose, content and location of each of the mandatory files.

6.6.1 Setup Files

1.a.1.7. urls.json

The *urls.json* file contains the server addresses to download Meta data from the JnJ digital library.

The *urls.WellDataURL* is used to download data from PLTS.

The file should be in c:/windows/urls/urls.json

Content

```
{"EFSURL": "http://xx/",
"PlateDataURL":"http://xx/",
"WellDataURL":"http://xx/",
"NeuroCellFactoryURL": http://xx/}
```

6.6.2 Input Files

1.a.1.8. barcode.txt

This file should be stored next to the tiff files. The barcode number should be the first item in the file. The rest of the file is omitted but can be used for comments or other numbers like. E.g. Orginal PLTS number. EFS number, for human reference

Content

1400035

1.a.1.9. *_Analysis.xml

The * is the name of the analysis.

Content

```
<Name>Frequency (Hz)</Name>
<Val>0.2500000000000000</Val>
<info>The frequency with which the temporal pattern is repeated</info>

<Name>Delay Time (ms)</Name>
<Val>6000.002070</Val>
<info> Delay from the beginning of the movie to the start of the temporal window of interest.
</info>

<Name>Pulse count</Name>
<Val>1</Val>
<info> Number of temporal repeats to be averaged out</info>

<Name>Partial Frequency (Hz)</Name>
<Val>0.2500000000000000</Val>
<info> The stimulation frequency during burst </info>

<Name>Partial Delay Time (ms)</Name>
<Val>0.0</Val>
<info> Delay from start window to start burst stimulation </info>

<Name>Partial Pulse count</Name>
<Val>1</Val>
<info> The number of spikes in the spike (burst) train</info>

<Name>Eigenvalue Number</Name>
<Val>0</Val>
<info>Selection from the eigenvector to create mask from, choices: 1 or 2 or 3 or 4, For
interactive selection during processing choose: 0. </info>

<Name>Reuse Mask</Name>
```

```

<Val>1</Val>
<info>choice: 0 or 1, create new mask = 0 or reuse existing mask = 1 </info>

<Name>Photo Bleaching</Name>
<Val>linInt</Val>
<info>choices: linInt or 2expInt</info>

<Name>Frame Selection</Name>
<Val>(1:end)</Val>
<info>Selection of the frames of the movie you want to use for mask creation. Temporal window must be inside the frame selection</info>

```

Camera Exposure Time (s) - 0.03000

<info>The framerate with which the frames were captured.</info>

6.6.3 Output Files

1.a.1.10. *_analysis.txt

Well features:

- peak amplitude is the average amplitude of all the synapses in the well.
- MstdSR: is the standard deviation of the average synapse response.
- MiASR: is the frame where the maximum average response was found.
- SizeWeightedMASR: is the size weighted Maximum Averaged Synapse Response
- SwmiASR: is the frame where this maxim is found
- Fps: frames per second is the image acquisition frequency
- UpHalfTime: Time where 50% of the amplitude was reached.
- DownHalfTime: Time where 50% of the amplitude was reached, going down.
- Tau1: is the time constant of the exponential fitted curve (decay).
- AmpSS: is the amplitude of the exponential curve
- NSynapses: is the number of synapses found in the well.
- AUC: is the average area under the curve for each well, using only positive values
- NAUC: Is the negative Area under the curve, is the sum of all negative values in the window of interest.
- Tau1PA: The time constant of the fitted curve on the pixel average
- AmpPA: the amplitude of the pixel average
- TOPA:
- Error: indicates if error was thrown when processing this well

Example Content:

peakAm p	mstdSR	miASR	sizeWeightedMASR	swmiAS R	fps	UpHalfTi me	downHal fTime	tau1
1.54043 8728	0.78671 2257	41	1.47314 4707	67	32.8083 9895	1.20200 2313	1.23764 3876	0.03368 3578

ampSS	nSynapse s	AUC	nAUC	tau1PA	ampPA	t0PA	error
1.540438 728	9	3.154051 881	- 2.151604 235	0.038636 133	0.235747 761	0.001650 434	0

1.a.1.11. *_traces.csv

- Different traces averaged over the well. Pixel average, synapse average and a size weighted synapse average trace. A raw trace, with no ΔF/F normalization. And for now, obsoleted: AverageResponse1

Example Content:

time	SynapseAver age	PixelAverage	SWSynapseA verage	rawAverageR esponse	AverageResp onse1
0.03048	0.021353	0.000342	0.049423	108.5245	0
0.06096	-0.05501	0.0047	-0.05502	108.8698	0
0.09144	0.04362	0.002033	0.015355	108.452	0
0.12192	0.049183	-0.00843	0.0562	107.1875	0
0.1524	0.043187	-0.00379	0.037258	107.5639	0
0.18288	0.152516	-0.00847	0.078719	106.9271	0
0.21336	-0.06052	-0.01554	-0.03935	106.0308	0

1.a.1.12. *_PPsynapses.txt

- synapseNbr : each synapse is numbered and has a unique number in the mask.
- synapseAmplitude1: The amplitude of the first spike
- synapseMaxFrame1: The frame in the partial window in which the max of the spike happened,
- synapseSize: the number of pixels a synapse has
- noiseSTD1: A rough noise estimate of the signal, based on a few first samples

- aboveThreshold1: If the signal is bigger than a few times the noise
- UpHalfTime1: time when 50% of amplitude is reached
- downHalfTime1: time when 50% of the amplitude is reached, going down
- tau_1: The time constant of the exponential fitted
- ampSS_1: The steady state amplitude, or the amplitude of the fitted exponential.
- error_1: defines if an error occurred during processing of this synapse.
- xCentPos: the x center value in pixels of the identified synapse
- yCentPos: the y center value in pixels of the identified synapse
- bboxUx: the x value of the upper left corner of the bounding box of the synapse in pixels.
- bboxUy: the y value of the upper left corner of the bounding box of the synapse in pixels.
- bboxDx: the x value of the lower right corner of the bounding box of the synapse in pixels.
- bboxDy: the y value of the lower right corner of the bounding box of the synapse in pixels.
- AUC_1: The area under the curve of the first spike of only the positive values.
- nAUC_1: The negative area under the curve of the first spike, normally this should be close to zero, or much less than the AUC_1, which is the area under the curve of all the positive values.

Example Content:

synapse Nbr	synapse Amplitu de1	synapse MaxFra me1	synapse Size	noiseST D1	aboveTh rehold1	UpHalfTi me1	downHal fTime1	tau_1
1	3.276196	41	4	0.510576	1	0	0	0.009066
2	0.820757	41	5	0.142524	1	0	0	0.030628
3	0.970312	41	5	0.115536	1	0	0	0.029557

ampSS _1	error_1	xCentP os	yCentP os	bboxUx	bboxUy	bboxDx	bboxDy	AUC_1	nAUC_ 1
3.276196	0	3	1.25	4	2	2	1	11.90844	-10.8646
0.820757	0	114	45	115	46	113	44	6.159828	-4.27288
0.970312	0	158	98	159	99	157	97	3.368342	-5.63795

1.a.1.13.

*_RawSynTraces.csv

Time is seconds and the raw intensity data of each identified synapse. The numbering is the same as the numbering of the features.

Example Content:

time	syn1	syn2	syn3	syn4
0.03048	125.3956	286.5456	355.3456	214.8123
0.06096	93.48167	289.5317	346.5317	214.3983
0.09144	117.5156	328.4656	302.8656	268.8211
0.12192	196.1576	351.0076	334.8076	201.4076

1.a.1.14.

*_synapses.txt

- MaxSyn: The maximum of the synapse
- miSyn : The frame the maximum was reached
- synapseSize : The size of the synapses
- noiseSTD : A noise estimate of the synapse response
- aboveThreshold : 0 or 1 indicating if a threshold is reached
- UpHalfTime : Time when 50% of the amplitude is found
- downHalfTime : Time when 50% of the amplitude was found, when trace going down
- tau1: Time constant of the fitted exponential curve
- ampSS : Amplitude of the fitted exponential curve

Example Content:

Columns 1-9

maxSyn	miSyn	synapseSize	noiseSTD	aboveThreshold	UpHalfTime	downHalfTime	tau1	ampSS
3.27619 6417	41	4	0.44566 7319	1	0	0	0.00906 5867	3.27619 6417
0.82075 7398	41	5	0.19621 901	1	0	0	0.03062 8283	0.82075 7398
0.97031 1923	41	5	0.13245 559	1	0	0	0.02955 711	0.97031 1923
1.97237 9056	41	9	0.15113 844	1	0	0	0.03420 7874	1.97237 9056

Columns 10-19

error	xCentP os	yCentP os	synaps eNbr	bboxUx	bboxUy	bboxDx	bboxDy	AUC	nAUC
0	3	1.25	1	4	2	2	1	11.908 44085	- 10.864 62221
0	114	45	2	115	46	113	44	6.1598 28159	- 4.2728 75684
0	158	98	3	159	99	157	97	3.3683 41905	- 5.6379 53351
0	297	306	4	299	307	295	305	6.6307 71327	- 2.4186 48581

1.a.1.15. *_synTraces.csv

- The time is in seconds and the trace of each synapse is the ΔF/F intensity.

Example Content:

time	syn1	syn2	syn3	syn4
0.03048	0.122152	-0.04859	0.063036	-0.07489
0.06096	-0.15861	-0.03763	0.038751	-0.07477
0.09144	0.059588	0.092547	-0.0896	0.161024
0.12192	0.764785	0.168363	0.007808	-0.12681
0.1524	0.581206	0.002535	-0.23654	0.08011

6.6.4 Log Files

1.a.1.16. Process_*.tif.txt

This file contains a log of the processing done on the file. Ranging from processing time, S3T version number, the analyses which where run, and the different settings for each analysis used.

Example Content:

S3T: V0.71-Rel

start: 2018-10-25 3:45:26.486

start Analysis: 01AP_1st1AP_Analysis.xml: 2018-10-25 3:46:33.014

```

processed: 2018-10-25 3:48:9.587
settings:
  stimFreq = 0.5
  Number of stimuli =1
  OnOffset = 26
  Partial stimFreq = 0.5
  Partial number of stimuli =1
  Partial OnOffset = 0
  Eigen Value Nr = 1
  fps = 32.8084
  dt = 0.03048
  Reuse Mask = 0
  Threshold = 0.025518
  data Frames Selection = ([1:98])

start Analysis: 01AP_2nd1AP_Analysis.xml: 2018-10-25 3:48:9.59
processed: 2018-10-25 3:49:42.535
settings:
  stimFreq = 0.5
  Number of stimuli =1
  OnOffset = 16
  Partial stimFreq = 0.5
  Partial number of stimuli =1
  Partial OnOffset = 0
  Eigen Value Nr = 1
  fps = 32.8084
  dt = 0.03048
  Reuse Mask = 0
  Threshold = 0.023968
  data Frames Selection = ([330:422])

```

1.a.1.17. processSoft_*.zip

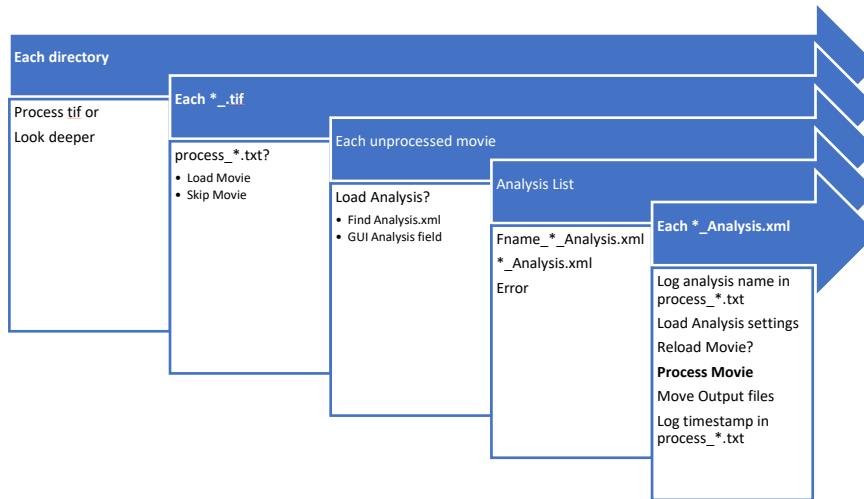
If a directory is finished processing, a copy of the software's source code is saved in the `./process/` subdirectory. This gives full back-traceability on the processing details of the software. The naming of the *.zip contains the computer and username which ran the analysis.

e.g.: **processSoft_computername_username.zip**

When a directory is rerun by the same computer by the same user, the zip file is overwritten. When multiple users and/or computers ran the same directory. Each computer/user will add its version. Currently it is not logged for each file, which computer/user processed the data.

6.7 Multi experiment batch processing

In overview, the file management framework to process multiple directories with multiple *.tif files with different types of analyses, can be summarized by the following picture. Each part is explained in more detail in the following sections.



6.7.1 File processing in multiple directories

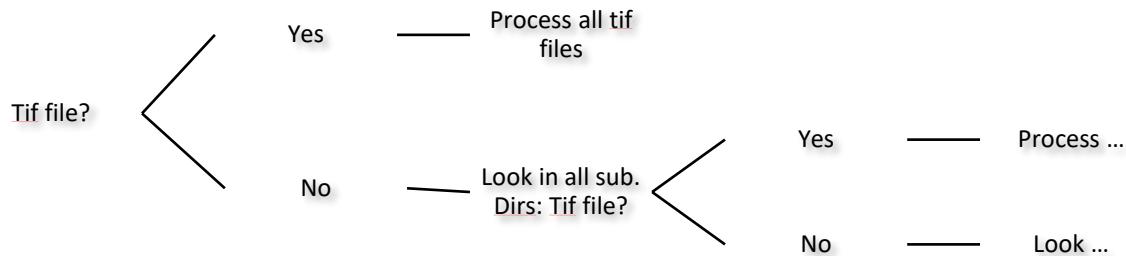


Figure: godeep mechanism to traverse directories and subdirectories

The way the files are processed by one or more computers is defined by the *godeep* mechanism. This mechanism will look in the defined directory for the presence of a *.tif file. If there is one or more *.tif files, it will process all of them in that directory alphabetically and stop like explained in the next section. Only if there is no *.tif file in the directory, it will look in all the direct subdirectories of that directory to find and process them. This is done recursively (up to 6 levels deep). The strength of this mechanism is that it allows to move files, which should not be processed, temporally in a subdir of their original location: If a *.tif file remains, the files in this subdirectory will not be processed.

6.7.2 Processing a directory with *.tif files

If a directory is processed, a ./process/ subdirectory is created.(if not existing already)

For each *.tif file the processing starts, a **process_filename.txt** file in the ./process/ subdirectory is created. This process file contains a.o.: the version number of the S3T software processing and the date and time the processing started. When processing of the file is finished, the completion date and time is added together with the S3T: analysis configuration settings at the end of the processing.

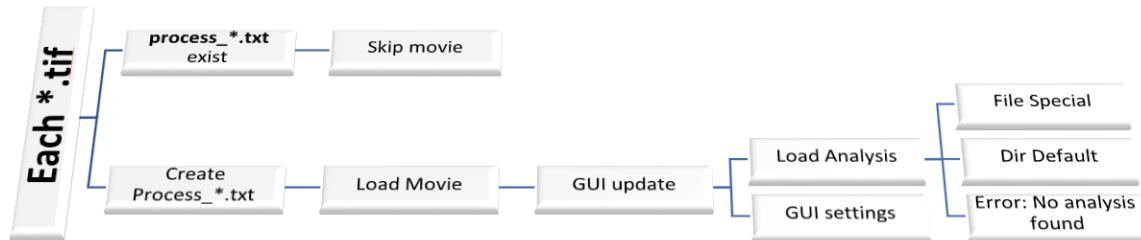


Figure...

When the processing of a file is started, the **process/process_filename.txt** file is checked for existence. If it does, this means that another instance of S3T (on the same or other computer) is already processing (or has processed) this file. So S3T will skip this file and try to process the next one.

If the file is not processing (or processed), S3T will create a new log file and load the *.tif file. During loading, every 100 frames loaded, a frame of the movie is shown.

If loading is successful, the filename in the top-left corner of the GUI is updated.

6.7.3 Loading Analysis.xml

Then: If the load analysis checkbox is checked (default), S3T will generate for the *.tif file a list of analyses to be run.

- First it will search in the directory for all similar named analysis file(s): **fname.tif_*_Analysis.xml** (special Analysis)
- If none, a search in the directory for normal analysis file(s) is done: ***_Analysis.xml** for which * has no '.tif' in it (directory default Analysis)
- If no analysis files are found, an error is put in the process file:

Sorry could not find analysis file.
.

For each *.tif file found, this list of analyses is than processed one by one in alphabetical order.

1.a.1.18. Processing analysis files

Processing of an analysis on a movie consist of:

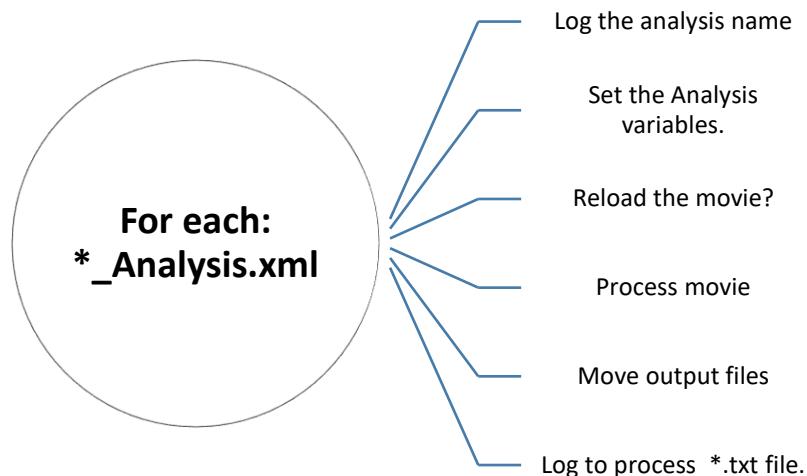


Figure..

- 1) Writing the analysis name in the ./process/process_*.txt log file.
- 2) Loading and setting the Analysis configuration variables from the Analysis.xml files into the S3T software.
- 3) If the analysis requires a reload of the movie, the movie is loaded
- 4) The movie is processed,
output data files are stored in the ./output/ directory. Mask and debug pictures are stored next to *.tif file
- 5) The analysis debug/result files are moved from the *.tif file directory and ./output/ directory to their respectively named ./*_Analysis/ directory and ./*_Analysis/output/ subdirectory. A copy of the mask is placed next to the *.tif file to allow automatic mask reload of the newly created mask.
- 6) Process log data is appended to the /process/process_*.txt log file.

If load Analysis is not checked,

- 1) The analysis configuration values in the S3T GUI are used.
- 2) The movie is processed,
- 3) Process log data is appended to the /process/process_*.txt file.

6.8 Movie processing

6.8.1 Overview

The movie processing consists of 3 main parts: Frame selection, Mask creation (or mask reload) and Signal processing. Each part is roughly broken down into sub-parts as defined in the next figure.

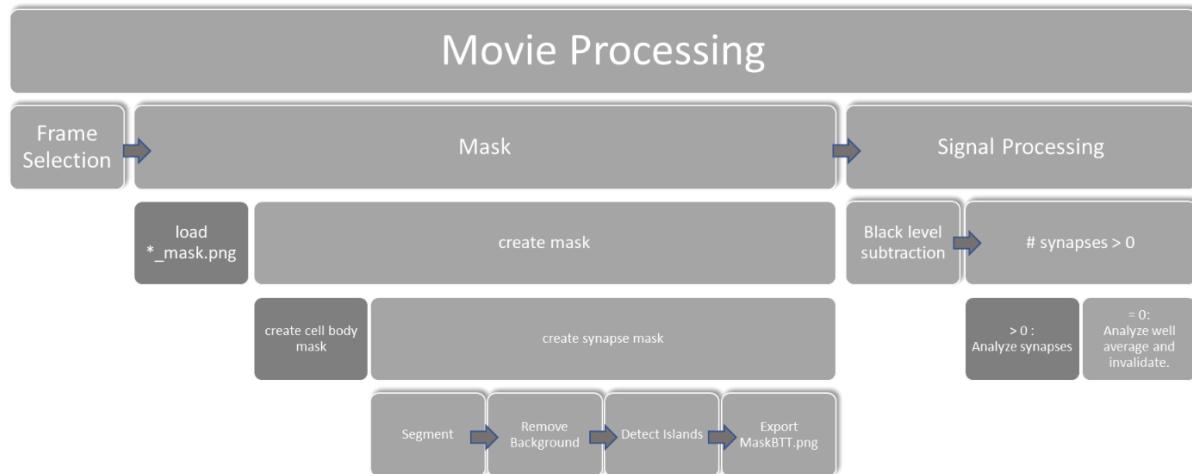


Figure: High Level overview of the Movie Processing script. 1) Selection of the frames of interest as defined in the analysis file. 2) Mask load or mask creation, a) If mask reload, => load *_mask.png in directory, b) If not mask reload => create mask. 1) If Cell Bodies => create cell body mask, 2) If Synapses(default) => create synapse mask: Segment, Remove Background, Detect Islands, Export MaskBTT.png. 3) Signal Processing: a) Black level subtraction, b) If number of synapses > 0 => analyse synapses c) If no synapses found => analyze well average and invalidate results.

6.8.2 Frame selection

The frame selection happens based on the selection of the frames of interest as defined in the analysis configuration file. The other frames are removed from memory. When they are needed in another analysis, a reload of the movie should happen as can be defined in the analysis configuration file. In the case that all data needed for the next analysis is available, a reload of the movie can be skipped, resulting in a considerable time gain. The default is set to (1:end). Since this text field is evaluated by S3T (Matlab), mathematical expressions to calculate the frames are allowed here. E.g. (1:floor(0.75*33)).

6.8.3 Mask

1.a.1.19. Mask loading

A mask can be created or a mask can be automatically reused by setting the reuse mask flag in the analysis configuration file. The mask will load the ***fname_mask.png*** residing next to the ***fname.tif*** file. Each time a mask is created, the mask will be stored next to the ****.tif*** file. Also a copy of the mask is copied to the analysis output directory.

1.a.1.20. Mask creation

If no mask is available or a different mask with different settings is desired, a new mask can be created.

The mask can be a synapse mask (default) or a cell body mask (limited).

The Mask creation is done in multiple steps:

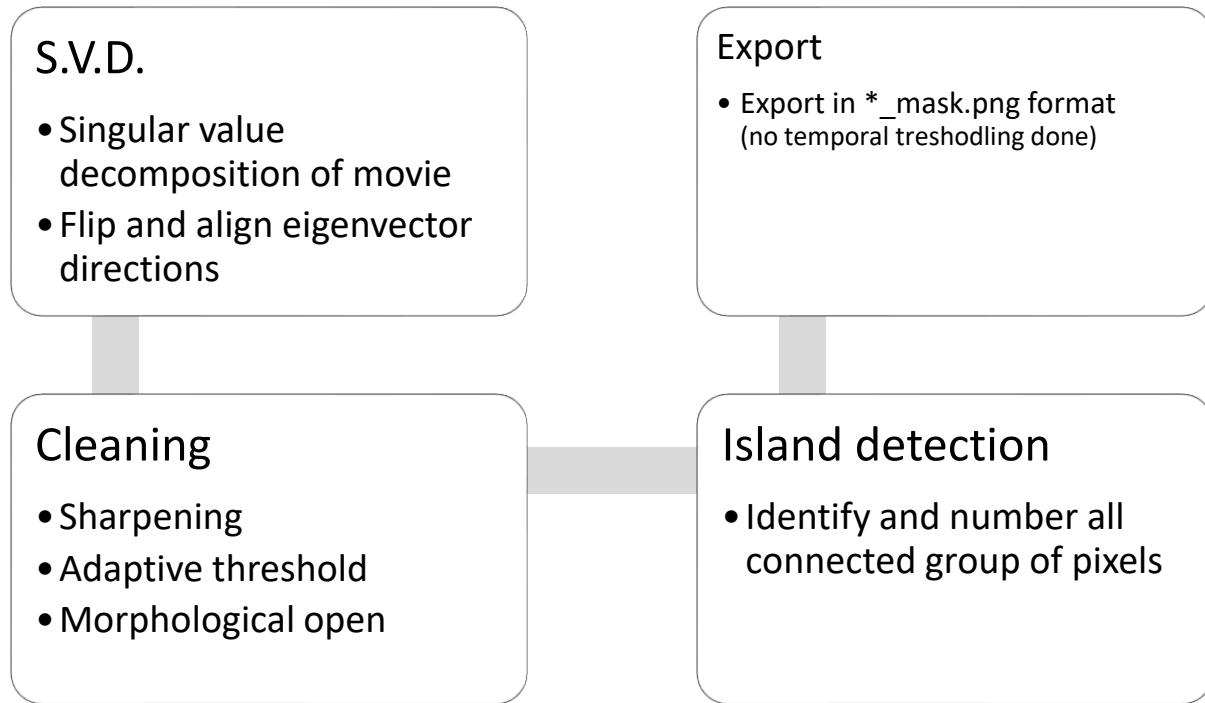


Figure: Overview of the mask generation algorithm. 1) The SVD (singular value decomposition) of the movie is calculated, 2) Image processing, Thresholding and cleaning is done, 3) Pixel island are detected and numbered, 4) The initial mask is exported for reuse or quality control.

For more information regarding each section see further.

6.8.4 Signal processing

Signal processing can be done on many levels. Ranging from individual fluorescent responses of a single synapse (4) to temporal averages and (synapse) spatial averages. These averages are interesting to acquire more noise robust features. It should be noted that with spatio-temporal averaging, also the synapse variability is getting lost. This is certainly an interesting feature but is difficult to acquire correctly due to different noise sources. For this, to get the most information out of the data, the data should be analysed at the most granular level the signal to noise ratio allows. From there, multiscale statistics can be generated to analyse compound effects on different spatio-temporal resolutions.

Signal processing overview scheme:

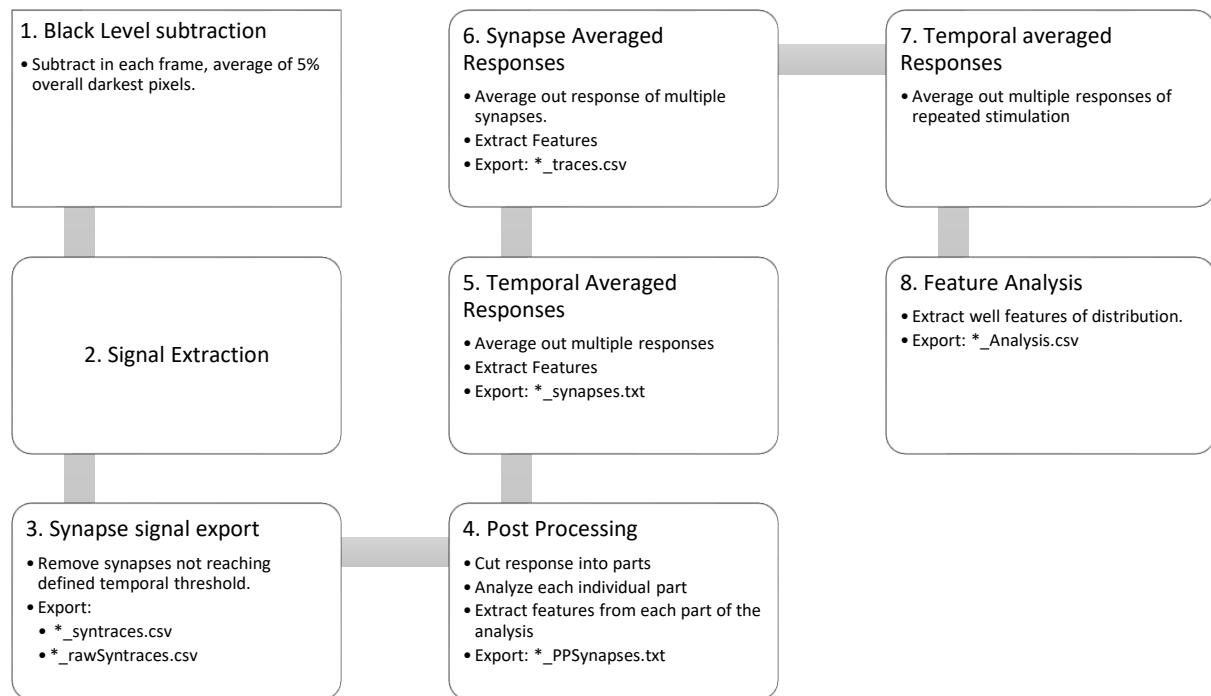


Figure: The signal processing consists of: 1) Black level subtraction, 2) Signal extraction, 2b) Export of signal plot (optional), 3) Export Synapse signals, 4) Postprocessing, 5) per synapse temporal averaging 6) Synapses averaging, 7) Temporal averaging, 8) Average signal Analysis

6.8.5 Mask creation details

1.a.1.21. Singular Value Decomposition (SVD)

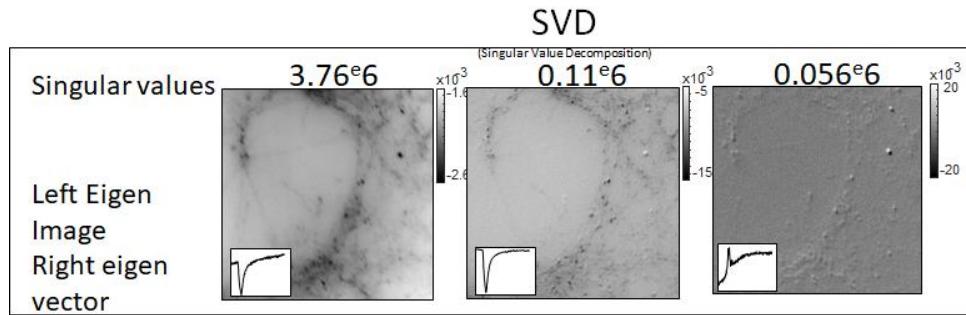
For each analysis of a *.tif movie a new mask can be created or the mask from previous analysis can be reused. This by specifying the reuse mask field in the analysis configuration file.

If a new mask is created, the mask created on the frames defined in the frame selection field.

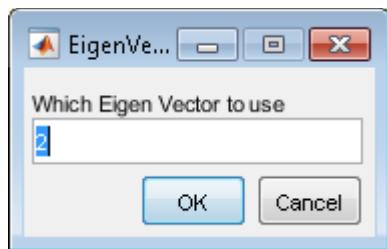
This frame stack is reshaped into a matrix with each column containing all the pixels of a single frame. The width of the matrix is equal to the number of frames.

From this matrix an ‘economic’ singular Value Decomposition (SVD) is calculated. So, for each singular value of the decomposition we have a left and right eigenvector. The left eigenvector represents all the pixels of the frame. And their contribution to that singular value. The right eigenvector contains the temporal behavior associated with that singular value. The singular value itself is related to the amount this combination of left – right eigenvector can explain the dynamics in the movie. We assume that the singular values are sorted from big to small, so that the first left and right eigenvector are associated with the largest singular value.

Typically, the first left and right eigenvector explain something very similar to the averaged response over space and time. The second eigenvector explains the variance between this averaged behavior and is most of the time the most interesting left-right eigenvector to explain the variance in the movie produced by the synapse behavior. For this the default eigenvalue which is chosen is to create the synapse mask from is 2. For some experiments: primarily defined by the combination of fluorescent reporters and stimulation pattern other eigenvectors might be more interesting and can be selected by the eigenvalue number field in the analysis configuration file.



The left and right eigenvectors are displayed during processing. If the best eigenvector to use is still unknown, the eigenvector can be set to 0 in the analysis configuration file and the eigenvalue can be chose interactively in a pop-up window.



SVD Sign alignment

When the eigen vector is chosen, the sign of the eigen image is first standardized to make synapses white (positive values) and background black (negative values). (This because the SVD can also produce a mirrored decomposition). The sign is defined by counting the number of black pixels vs white pixels. Black: $\text{pixel} < \text{mean(image)}$ and white: $\text{pixel} > \text{mean(image)}$. If the number of white pixels is more than the number of black pixels the image is inverted.

1.a.1.22. Thresholding

First a sharpening filter is applied: Matlab: `imsharpen('Radius',16,'Amount',40);`

Depending on the radio button selected a 2 sigma or 3 sigma threshold is set. The sigma is the standard deviation of the selected eigen vector image after sharpening.

Another option exists to enter a fixed threshold value.

After thresholding, (everything above the threshold = 1, below = 0) cleaning of the mask is done with a 1-pixel morphological opening (= erosion + dilation) to remove noise speckles.

This is the initial mask B/W which will be used to extract the signals.

1.a.1.23. Island detection and numbering

Groups of connected pixels (synapses) are identified and given a unique number. Features of the synapses are calculated as: synapse center, bounding box, synapse size.

1.a.1.24. Export MaskBTT

The maskBTT (Before Temporal Threshold) is exported into a 16-bit gray value ***_maskBTT.png** picture. The background is black (=0), with each synapse pixel is encoded as 2^{16-j} , with j the synapse ID number. (j==1 for the first synapse). This allows to have each time the same synapse numbering and each synapse can be retrieved by its number.

6.8.6 Signal Processing details

1.a.1.25. Background subtraction

The black of the movie (= 0) is defined by a search for the 5% darkest pixels when averaged over all frames. The average intensity of these pixels is calculated in each frame and this value is subtracted in each frame. This to make the black level more uniform over multiple experiments.

1.a.1.26. Signal extraction

The islands (synapses) are numbered and for each island (a group of connected pixels) the average dynamic response is calculated.

If no synapses are found, a warning is displayed, and the average of the entire window is calculated.

SignalPlot (Optional)

- For each well a *_signal.png picture is generated which illustrates the $\Delta F/F$ of all the synapses.

Also, for each well a *_signalAT.png is created which is the same image but with the synapses not reaching the temporal threshold value removed. The threshold can be set to a fixed value or a 2, or 3 times the noise standard deviation. This noise standard deviation is calculated on the first 10 samples of the temporal window.

1.a.1.27. Export synapse signals

temporal threshold

The delta f over f is calculated for each synapse signal. Then for each response a noise estimate is done (standard deviation of first 10 samples in the window) and the maximum signal during this window of frames is compared to get a signal noise estimate. If the signal maximum does not reach the defined temporal threshold, the synapse is removed from the synapse signals table and also from the mask.

1.a.1.28. Export Mask

The mask is exported into a 16-bit gray value *_mask.png picture. The background is black (=0), with each synapse pixel is encoded as 2^{16-j} , with j the synapse identifier. j==1 for the first synapse. This allows to have each time the same synapse numbering and each synapse can be retrieved by its number.

1.a.1.29. Partial Processing (PP)

Partial processing is done especially for burst stimulus.

The delta f over f is calculated. If in the analysis the pulse count is set >1, multiple parts of the response are averaged out first. Otherwise, a temporal window with size related to the stimulus frequency parameter is selected.

This window is then partially processed. Meaning for all partial pulse counts, the amplitude , AUC, noise, ... are calculated and stored in the *_PPSynapse.txt files. (See section on *_PPSynapse.txt file for more information). If enough samples are available (>4), then an exponential fit is done on the decay of the of the synapse response.

Different warnings are produced here when timing information in the analysis configuration file are inconsistent with other parameters or the data.

1.a.1.30. Analyze single synapse signals

Here the DFF is calculated of all the synapse signals and then the synapses responses are averaged out when the pulsecount in the Analysis configuration file is >1. (See MR1 for more information)

Here all the features of the *_Synapses.txt file are calculated. (See *_Synapses.txt for more information)

The noiseSTD is calculated on the first 5 samples of the temporal window.

The above Threshold flag is set when the signal is > 2 noiseSTD.

1.a.1.31. Average Synapse Response

The DFF of all the synapse signals is calculated. From this the average is calculated and stored in *_traces.csv. A variation of this signal is calculated by calculating the average weighted by the synapse size. Also, the pixel average of the raw data, is calculated and stored in the rAR column.

The DFF of the rAR is stored in the AR column.

(See also *_traces.csv for more information)

1.a.1.32. Temporal averaging

The spatial average AR and ASR are temporally averaged with MR1. When the analysis configuration file has pulsecount> 1 , the temporal windows are averaged out.

1.a.1.33. Analyze average Response

The maximum of the response is searched for in the time window. The maximum is stored in the mASR. The frame in which this happens in the window is stored in miASR. Similar for AR the maximum and frame are stored in mAR and miAR. Similar for the synapse size weighted average response, swASR. The number of synapses found and if any errors were found during processing.

The variables are stored in the *_Analysis.txt files. For a detailed overview of all the variables see *_Analysis.txt). For each well a *_Analysis.png figure is also created, annotating some of the extracted features on the average synapse response.

6.9 Algorithm details

6.9.1 DFF Calculation

The delta f over f calculation, relates the current fluorescent intensity F with the base fluorescent intensity f0. $Dff = (F-f0)/f0$. During the DFF calculation, also a Photo bleach correction is done.

This photobleaching correction accounts for the diminishing fluorescence over time of the fluorescent markers. Based on the setting in the analysis configuration tool, a linear or double exponential photo bleaching can be done.

1.a.1.34. Linear Bleach correction

- A linear fit is calculated between the average of the first 3 samples and the average of the last 3 samples in the selected temporal window. The $\Delta F/F$ signal at point t is calculated as:

$\Delta F/F(t) = (F(t) - (F_0 + BC(t)))/(F_0)$, with F_0 the mean of the first 3 samples. $BC(t) = (F(\text{end}) - F(\text{start})) * (t - t(\text{start}))/(\text{t}(\text{end}) - \text{t}(\text{start}))$.

The advantage of this method is that it only needs 2 points start and end and that a linear fit maximum error is bounded since it is a pure interpolation. More points can be used to make it more robust against noise. If the time window is small, a linear approximation of the photobleaching is certainly a good, fast and robust approximation. When the interval is longer, more complex fits are recommended.

1.a.1.35. 2xExponential Bleach correction

When the synapse signal does not reach the base fluorescence for a longer time, a double exponential curve fit might be capable of modelling the photobleaching effect more accurately than a linear fit.

$y = a \exp(b x) + c \exp(d x) + d$

A double exponential curve is fitted on the concatenation of the first and last 30 samples of the temporal window under investigation.

This is done by solving a linear set of equations. The system to solve is sometimes ill conditioned and gives warnings about this during processing. This might result in a bad photo bleaching fit for that synapse.

1.a.1.36. Auto2expInt correction

This algorithm tries to find iteratively find out which samples in the trace are part of the baseline, and which frames are not, since some activity is happening at that moment. This selection of points is used to fit the 2exponential on for photo bleach correction.

1.a.1.37. Autolinint correction

This algorithm tries to find iteratively find out which samples in the trace are part of the baseline, and which frames are not, since some activity is happening at that moment. This selection of points is used to fit a linear line for photo bleach correction.

1.a.1.38. Global correction

This starts with the calculation of the average response of the field of view over time. In the response the algorithm searches for the frames where the average is at baseline level. On these frames, a 2exponential curve is fitted. The values of the curve for each frame are then used to calculate the ratio-metric factor the intensity is multiplied with, to arrive at the photo corrected stack of frames. The ratio-metric factor for a frame, is the ratio of the average intensity at the first frame with average intensity at that frame. In which the intensity for that frame is extracted from the 2exp interpolation.

1.a.1.39. LoadPBProfile

When this option is set, a profile is loaded from the PBprofile.dlm file in the installation directory. This file contains a ratiometric curve. The intensity of each pixel is multiplied by the factor for that frame.

6.9.2 MR1 (Multi Response to 1)

MR1 is the function which is used to calculate the temporal averages. When multiple similar stimulations are given, it might be interesting to align and average out these multiple responses to obtain a less noisy synapse response. This allows for more accurate fits of amplitude, decay rate ...

The alignment is done based on the numbers provided in the analysis configuration file. The inter-stimulation period (ISP) is calculated based on the stimulation frequency (stimFreq) and the imaging frequency (fps).

$$\text{ISP} = 1/\text{stimFreq} * \text{fps}$$

The different parts are selected out of the data by:

```
part(:,I)=data(OnOffset+floor((I-1)*ISP)+(1:dCOF));
```

The onOffset is defined by the delay in the analysis configuration file (After conversion from ms to frames).

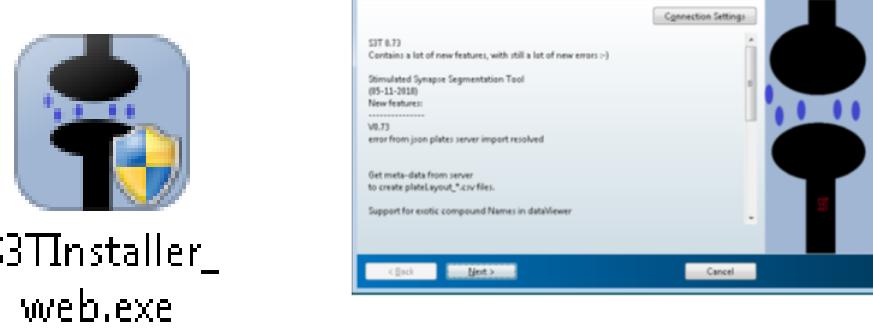
$$dCOF = \text{floor}(ISP)$$

These different parts of the full response are then averaged out.

6.10 Installation

Run the installer from)

<https://github.com/meChiel/S3T/releases>



6.11 Errors and bad analysis

What to do when errors happen:

6.11.1 Errors during processing

If during processing of the *.tif files, an error happens which aborts the further processing of the files. More info can be found in:

- 1 If no changes are shown in the GUI and/or the processor in the task manager is also not calculating heavily. Look at the command window. If an error occurred: an error message must be displayed here. This could give you also some hints of the type of error.
- 2 In the upper left part of the GUI, the last correctly loaded file is displayed. This file is probably under processing. Make sure there is nothing wrong with this file.
- 3 Look in the process folder. For each file processed a process log file is created which can give you the information on which file and which analysis the processing failed.
- 4 To rerun the analysis on the last file: e.g. **fname.tif**, delete the **./process/process_fname.tif.txt** file. If NOT the file will be SKIPPED for processing as explained in Appendix 6.7.

6.11.2 Bad analysis

How to detect?

When the processing is done, some semi-automated quality checks can be executed by looking at the generated overview pictures or doing a particular *.png search in the folder with picture preview in explorer. This last option will give you the name of the file with bad analysis.

- 1) Overview_analysis.png or a *_Analysis.png search can be done to look for the quality of feature extraction. E.g. the photobleach correction. The quality of the exponential fit. Amplitude detection. AUC calculation, ...
- 2) Overview_align.png or a *_align.png search can be done, to look for the temporal window and the alignment of multiple temporal windows before the temporal mean response was calculated.
- 3) Overview_signal.png or a *_signals.png search shows the different synapse signals. The *_signalsAT.png gives an overview of the synapse after temporal threshold was done.
- 4) Overview_mask.png or a *_mask.png gives an overview of the different masks created. A *_maskBTT.png gives an overview of the masks before temporal thresholding was done.

How to resolve?

Changing the temporal window in the analysis configuration to making them fit the data more. Increase (partial) delay, increase (partial) frequency (= decrease window size). This resolves most of the analysis problems. A choice can be made for linear and 2x exponential photobleaching. Make sure the first and last samples in a temporal analysis window are non-stimulated, I.e. are at base fluorescence level.

If the mask is bad, the threshold can be changed with the radio buttons to a higher or lower value.

If this is not the reason, you might want to change the eigenvector image you started mask creation from. This can be done by setting the eigen value number field to another value than the default 2. Also, the eigenvalue number can be set to 0 for interactive eigenvector selection.

If only a single movie has bad analysis, a modification of the default analysis for that movie can be done by prepending the filename onto the Analysis.xml file: fname.tif_*_Analysis.xml.

One of the reasons, why a particular movie might have a different eigenvector sorting, is because a pixel errored during recording (hot pixel). The SVD based mask creation algorithm will recognize this event, (depending on the intensity value of the errored pixel). The errored pixel is then typically the single pixel shown in the mask. Correcting this bad pixel value in ImageJ (by replacing it by some neighboring pixel value in that particular frame), resolves this artifact and will lead to normal eigenvalue sorting and mask creation. Alternatively, the next eigenvalue can be selected for that particular movie as described above.

7 References

1. Dorostkar, Mario M, Elena Dreosti, Benjamin Odermatt, and Leon Lagnado. "Computational Processing of Optical Measurements of Neuronal and Synaptic Activity in Networks." *Journal of neuroscience methods* 188, no. 1 (2010): doi:10.1016/j.jneumeth.2010.01.033.