**AnalysisTools\_MWLab** Library of Programs (5/30/2019)

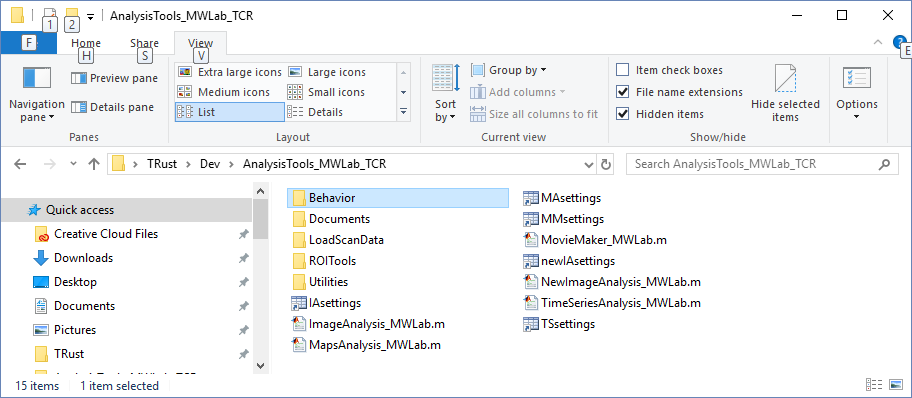
Home folder for source code currently located on Synology Disk Server:

\\155.98.15.105\software\analysis\_software\matlab\

Naming system: Analysis\_Tools\_MWLab\_year\_month\_day (last updated)

e.g. AnalysisTools\_MWLab\_2019\_05\_30\

Recommendations: Always copy the whole folder including subfolders – many programs depend on other programs. Lab standard is to put a copy on your local computer under User/Documents/MATLAB/. Add this folder including subfolders to your MATLAB path. To learn more about each program type “help <program>” in command window or better yet: “edit <program>.” (Nullius in Verba)



Note: Errors during GUI startup may be fixed by 1) close the GUI, 2) delete settings, 3) try again

*ImageAnalysis\_MWLab.m* (*IAsettings.mat*, saves settings for this program) – Graphic User Interface (GUI) for viewing raw data files collected by most of our imaging rigs (Scanbox, ScanImage, Neuroplex-InVivo, Prairie). In MATLab, enter “help ImageAnalysis\_MWLab” for more info. Notes:

1. Data stored in fighandle.UserData.IAdata (try it, type ”fighandle = gcf”)
2. Filters on left panel are only applied to the current frame, not carried over to right panel.
3. Selecting 2 or more images and then <load selected> creates an averaged image stack.
4. This program does not select different odors when <get stimulus timeframes> is clicked.

*NewImageAnalysis\_MWLab.m* (*newIAsettings.mat*) – new GUI designed to replace both ImageAnalysis and MovieMaker. Ideal for looking at delta F movies to pinpoint rapid changes in response to stimuli. This program also improves your ability to draw ROIs and make difference Maps wisely.

1. Load 1 image at a time
2. Make and save images & movies: mean Fluorescence, dF, %dF/F, or overlays
3. Select odor trials for difference maps or movies
4. Draw ROIs on any image or movie frame. Overlay ROIs and show traces.
5. Use preliminary results to run a batch processing script or add more files in another GUI

*MapsAnalysis\_MWLab.m* (*MAsettings.mat*) – GUI for creating/viewing/modifying/saving “difference maps” - snapshots of fluorescence or changes (dF, dF/F) created using stimulus signals such as odor on/off (useful for drawing ROIs). MapsData structure can be saved and loaded for repeated analyses.

1. MapsData – struct w/saved baseline and response images for each file/odor/trial
2. Fighandle.UserData.figdata{} – current figure(s) w/ map stack in figdata.im (and other stuff)
3. Image Processing steps are done in the sequence (top to bottom) shown in GUI
4. <Save Maps as Tiff> stores all information needed to recreate the map in the Tiff header. Use MATLAB function “imfinfo” to view the header (or >Image>Info in ImageJ).
5. <OdorRespFile> button creates “ORfile” which includes Maps and a matrix of the mean values in ROIs (row) vs Odor (column). Use /ROITools/ROIvsOdorGUI to analyze these files.

*MovieMaker\_MWLab.m* (*MMsettings.mat*) – GUI for making/saving movies of F, dF, dF/F, or an overlay of these along with time series plots of selected ROIs. This has some nice display features like a light that turns on during odor presentations and a text showing the time. Mostly for show!

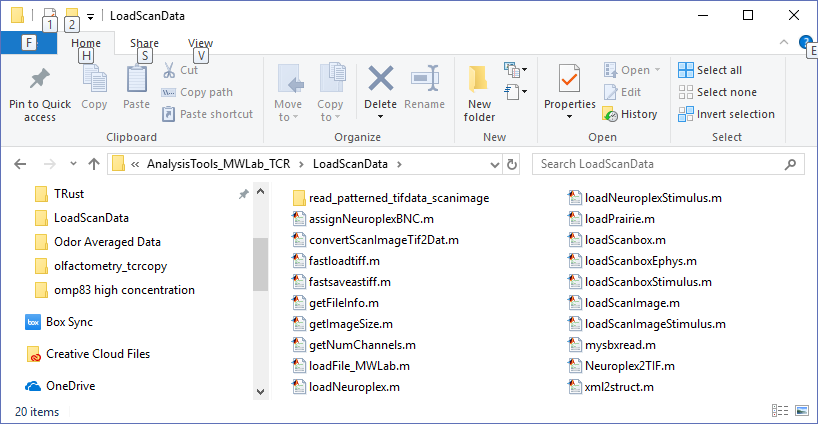
*TimeSeriesAnalysis\_MWLab* (*TSsettings.mat*) – GUI for making/viewing/saving time series data. This allows you to view full-length traces, or using auxiliary signals you may view individual odor trials. Also lets you do averaging of ROIs/trials/odors/files. TSdata can be saved and loaded for repeated analyses.

1. Fighandle.UserData.TSdata (“App Data”) full length time series
2. Fighandle.UserData.plotdata{} – data for current plot(s) - includes file().roi().odor().trials()
3. ROIs time-shifted based on centroid y-position
4. ROI’s “average” is a weighted average by #pixels (essentially one bigger ROI)
5. Select aux stimulus to get pre- to post- stimulus frames of odor trials (see getAllOdorTrials.m)
6. To get trials, TSdata interpolated to aux stimulus times (150Hz), then get frames for each trial
7. Low/High pass temporal filters, and then DF, DF/F calculations
8. Lastly, selected trials weighted equally in File/Odor/Trial averaging

**Folders:**

***/Documents/*** - This folder contains some program descriptions and info that may be useful for developers, including this document. Data structures are described briefly as well.

***/LoadScanData/*** – This folder contains programs for loading data of various types



*/LoadScanData/read\_patterned\_tifdata\_scanimage/* – this has programs provided by scanimage which enable faster reading of .tif files (These seem to do something weird with memory and are not used)

*loadFile\_MWLab.m* – general use program for loading all types of raw data into the command window. The output is a struct which can be called the “mwfile struct”. This program is used by all of the GUIs and calls other functions in this directory to load different file types. NOTE: updates to imaging rig software may affect the file formats and the loading of those file types!

Scanimage:

*loadScanImage.m* – loads raw data acquired using ScanImage rig and saved in .tif files (or converted to .dat/.mat format – see convertScanImageTif2Dat.m). It is important to know that ScanImage data are converted from int16 to uint16 by subtracting the median of the min values of the first 60 timeframes. Note: recently (5/28/2019) improved framerate estimates based on frame acquisition timestamp in the ImageDescription section of the tiff header.

*loadScanimageStimulus.m* – loads auxiliary signals saved in the scanimage .tif header

*convertScanImageTif2Dat.m* – converts .tif files to .dat/.mat (binary data and info files – like scanbox). This may be useful if you plan to open large scanimage files repeatedly – since tifs are very slow. However, a better choice may be converting to hdf5 format.

Scanbox:

*loadScanbox.m* – loads data acquired using Scanbox rig and stored in .sbx files (also uses sbx info files which have matching filename prefix without a file type extension)

*loadScanboxStimulus.m* – loads auxiliary signals stored in the sbx info files

*loadScanboxEphys.m* – load ephys data (odor, awake sniff, puff, etc.) saved in files with .ephys extension

*mysbxread.m* – a modified version of the sbxread.m taken for scanbox software. Allows us to read zero frames and just get file info faster.

Neuroplex (InVivo/scanimage):

*loadNeuroplex.m* – load neuroplex files (.da extension) acquired on in vivo rig or scanimage rig.

*loadNeuroplexStimulus.m* – load auxiliary signals stored in .da files using BNC inputs

*assignNeuroplexBNC.m* – required to assign BNC inputs to auxiliary outputs.

*Neuroplex2TIF.m* – converts Neuroplex to standard .tif format (to share data or view in other apps)

Prairie:

*loadPrairie.m* – reads .xml files from Prairie imaging rig (not used for years)

*xml2struct.m* – used by loadPrairie.m

Other:

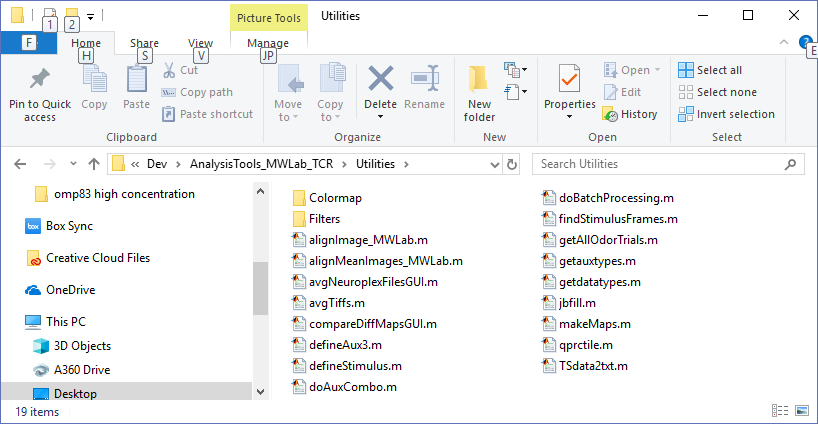
*fastloadtiff.m, fastsaveastiff.m* – used to read scanimage tifs, but also for loading/saving tif movies or large stacks of images.

*getFileInfo.m* – finds and displays metadata and .txt files for a selected file

*getImageSize.m* – determines files size (usually from header or metadata) without loading file.

*getNumChannels.m* – determines number of channels in file w/out loading file.

***/Utilities/*** – This folder contains many useful programs including some used by the analysis GUIs



*alignImage\_MWLab.m* – takes a matrix (stack of images), or mwfile struct (see loadFile\_MWLab), or no input (you are prompted to load a file) and aligns timeframes using intensity based, rigid image alignment (borrowed from Scanbox aligntool). Aligns everything inside a 20-pixel margin. Saves the results as a .align file (“-mat” file with m=mean image, T = shifts for each frame, and idx = index of frames to align).

*alignMeanImages\_MWLab.m* – This program aligns mean images of selected files. If a .align file exists the mean aligned file is used and results are “appended” to the current alignments, otherwise a .align file is created.

*avgNeuroplexFilesGUI.m* – GUI to average Neuroplex files by odor

*avgTiffs.m* – a simple program for averaging Tiff files and saving the resulting file.

*compareDiffMapsGUI.m* – GUI for comparing difference maps saved as .tif stacks

*defineAux3.m* – generate a simulated binary odor number encoding signal

*defineStimulus.m* – generate a simulated odor on/off signal

*doAuxCombo.m* – uses aux1(odor) and aux2(sniff) to find sniffs while odor is on

*doBatchProcessing.m* – a generalized script for aligning files, computing Maps Data, Time Series, etc.

*findStimulusFrames.m* – used by ImageAnalysis.m (old), finds timeframes for all odors using aux signal

*getAllOdorTrials.m* – used by newImageAnalysis, TimeSeriesAnalysis – finds all the valid odor trials and the indices of timeframes for a given aux stimulus and pre-post stimulus window both for image and auxilliary signals (e.g. allOdorTrials.odor().trial().auxindex, & allOdorTrials.odor().trial().imindex). These are used to make a list of available odor trials which may be selected to make plots, movies and such.

*getauxtypes.m* – returns a list of auxilliary signals which is used/shared by many other programs

*getdatatypes.m* – returns a list of datatypes (scanimage, scanbox, etc.) shared by many programs

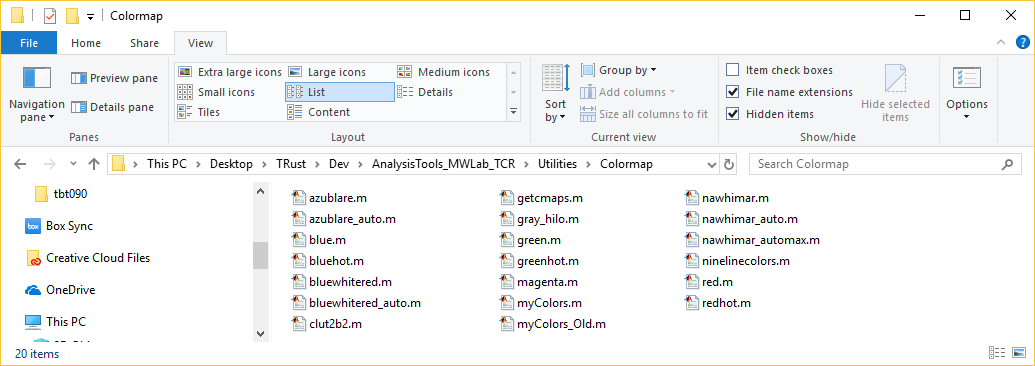
*Jbfill.m* – plotting tool used by TimeSeriesAnalysis

*makeMaps.m* – generates MapsData (used by MapsAnalysis\_MWLab.m) which includes baseline and response images for each odor trial based on the stimulus, baseTimes, and respTimes provided

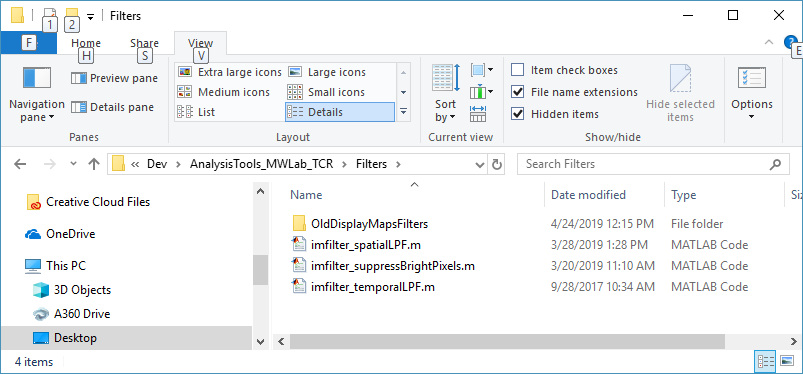
*qprctile.m* – gets percentiles (not exact, but may be faster than MATLAB’s prctile for integer data)

*TSdata2txt.m* – converts time series data structure (see TimeSeriesAnalysis\_MWLab.m) to text file

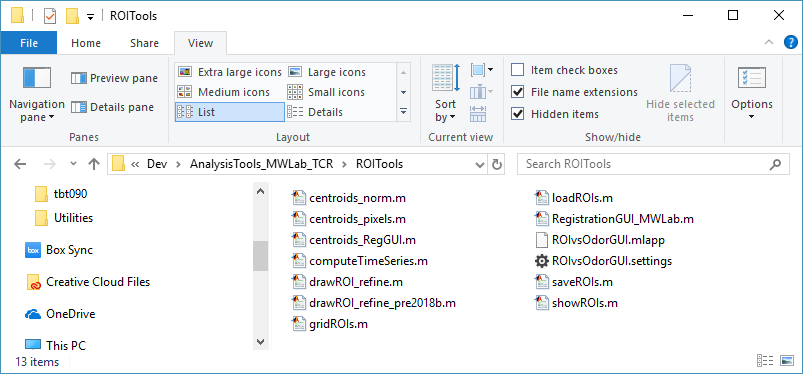
*/Utilities/Colormap/* – This folder contains functions for making color maps and line colors for plots/rois/etc. Edit the programs or try the color maps for more info. Note: *getcmaps.m* – returns a list of color maps used in the analysis GUIs



*Utilities/Filters/* - This folder contains programs for filtering images (includes some old custom filters)



***/ROITools/*** - This folder contains programs for creating/saving/loading ROIs, computing Time Series, co-registering images, and computing ROI centroids (with or w/out the registration results)



*centroids\_norm.m* – computes ROI centroids (positions are normalized 0-1 relative to image)

*centroids\_pixels.m* – computes ROI centroids (positions are in image pixels)

*centroids\_RegGUI.m* – computes ROI centroids using registration data (see RegistrationGUI\_MWLab.m; positions are in micrometers and relative to olfactory bulb anatomy)

*computeTimeSeries.m* – computes Time Series Data using ROIs and images in mwfile struct format (see loadFile\_MWLab for details). The resulting data are raw fluorescence (same as image) and are full-length time series not separated into odor trials.

*drawROI\_refine.m* – GUI for drawing ROIs, or adding additional rois. Some background images must be provided. Draw ROIs using polygon, quick squares, or intensity thresholding combined with region-growing via the scroll wheel on the mouse. Once ROIs are draw you can refine them using PCA or cross-correlation in a zoomed region. (*drawROI\_refine\_pre2018b.m* for earlier versions of MATLAB)

*gridROIs.m* – draw ROIs by defining a grid over the entire image (just a side project that’s not used)

*loadROIs.m*, *saveROIs.m* – load or save an ROIs file

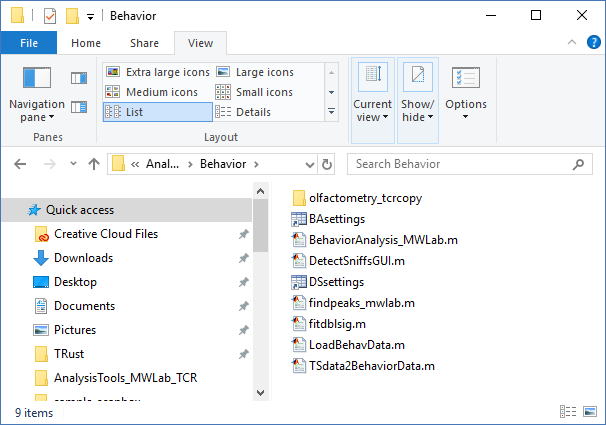
*showROIs.m* – display ROIs over a background image

*RegistrationGUI\_MWLab.m* – GUI for loading .tif images (usually from MapsAnalysis\_MWLab.m) and registering them to a global coordinate system representing the olfactory bulb. The registrations take into account pixel size and magnification from the different imaging rigs and resulting registration coordinates are saved as a struct in a .mat file with the file extension “.mwlab\_reg”. These results that be used to determine ROI centroids for comparison across different files and file types.

*ROIvsOdorGUI.mlapp, ROIvsOdorGUI.settings* – GUI program written using MATLAB “appdesigner” interface which reads “ORfiles” (Odor versus ROI). ORfiles can be generated using a button in MapsAnalysis\_MWLab.m or by other methods, and are essentially a saved .mat file with a struct called

ORdata, having fields {RefImage, RespMatrix, OdorList, ROIPos, Maps, Reg, MetaData}. Look in MapsAnalysis at the callback function named CB\_ORfile or edit this GUI for more info. Note: this program was recently developed and is still a work-in-progress.

***Behavior****/* - This folder contains programs for analyzing awake behavior data, such as natural sniffing in response to odor. Much of it is based on the older olfactometry program and is under development.



*/Behavior/olfactometry\_tcrcopy*/ - this folder is a copy of the old olfactometry program with some modifications to make it run in MATLAB 2018b – olfactometry.m is the main GUI program and most of the useful data processing is in programs named “\*\_extensions.m” (e.g. response\_fitting\_extension.m).

*BehaviorAnalysis\_MWLab.m ( BAsettings.mat)* – this GUI program is meant to replace the old olfactometry program with updates to accommodate our current data (it is also possible to convert our data to the olf class and use some parts of the olfactometry program, but this was the option we settled on). The underlying data is organized in the “behaviordata” struct (which can be created using a button in TimeSeriesAnalysis as long as ephys data is available; this is currently only set up on the scanbox rig).

*DetectSniffsGUI.m (DSsettings.mat)* – this program is used to find sniff inhalations and exhalations in the pressure signals acquired as part of the ephys data

*findpeaks\_mwlab.*m – modified version of the matlab function findpeaks.m from the signal processing toolbox. Used by DetectSniffsGUI.m

*LoadBehavData.m* – reads .dat files created using the NI Labview Behavior Programs used for awake animals (there are a few different versions of the behavior programs so you might need to modify this)

*TSdata2BehaviorData.m* – this converts saved TSdata (app data) into behavior data. The time series are separated into individual odor trials with ephys sniff signals for each “trial”.

“Nullius in Verba”

-don’t accept my descriptions, edit the files and figure out what they do for yourself!