Glue Sniffer ReadMe

Lagnado Lab

April 17

1 Load Movies

Loads your tiff file into the Igor workspace

2 Enter Params

Input the parameters for the Wiener filter here.

3 Split Channels

Splits the loaded tiff file into seperate channels (allows seperating stimulus from recorded channels).

3.1 Inputs and Outputs

Inputs :
your recorded tif file
Outputs :
_F : a flourescence data matrix
_stim : the stimulus file

4 Linescan

Converts the _F matrix into a useable format, a _L nTimePoints x nPixes linescan Matrix.

4.1 Inputs and Outputs

Inputs :
_F flourescence file
Outputs :
_L linescan matrix

5 New Define AZ

The new method of defining Active Zones. Begins by computing the temporal average of pixel intensity and plotting it. Users then place cursors around where they believe the center of a gaussian component lies, and Igor's curve fitting code attempts to decompose into a gaussian mixture model. Component means are constrained to lie within xPixels away from the placed cursors, and mixing coefficients A are constrained to be positive. This part can be a bit messy (due to Igor's curve fitting), but generally works pretty well. See appendix for notes on GMMs

5.1 Inputs and Outputs

Inputs :
_L : the data matrix
User cursors (graphically)
outputs :
_profile : temporal average
_SF : the nAZ x nPixel recovered spatial filters
_fit : the curve fit
_muHat : the estimated means
_sigmaHat : the estimated standard deviations
_aHat : the estimated mixing coefficients

6 New Temporal Profile

The new temporal profile method. Begins by taking the estiamted spatial filters _SF and spatially deconvolves the linescan matrix _L, extracting a nComponent x nTimePoints matrix _roiDatMat, of which each nComponent is a time series of ROI data (saved also as _AZ1). Each of these time series are then baseline corrected, deltaF over Fed, and a weiner deconvolution is used (the wiener filter made from the parameters from 'Enter Params"), outputting a deconvolved wave _D and and a smoothed wave _S.

6.1 Inputs and Outputs

Inputs :
_SF : the spatial filters estiamted from 'New Define AZs'
_L : the linescan matrix from 'Linescan'
tauRise and tau : wiener filter parameters from 'Enter Params'
Outputs' :
_AZ (cluster number) : the time series extracted by spatial deconvolution. Baseline corrected and dF/Fed
_AZ(n)_S : smoothed wave, after wiener deconvolution
_AZ(n)_D : deconvolved wave
_deconMat : matrix of deconvolved waves

7 Threshold

Graphical user interface for setting the threshold value required for deconvolved peaks to be considered events, defined as a local maxima above the set threshold. Plots the Deconvolved waves with a slider bar ranging from 0 to the maximum value of the deconvovled wave. Graphically displays which events are selected when slider bar is moved. Cycles through all AZs.

7.1 Inputs and Outputs

: **Inputs** :
_AZ(n)_D : deconvolved wave from 'New Temporal Profile' or 'Old Temporal Profile'
Outputs :
thresh : the threshold for finding events

8 Event Detection

Takes the deconvolved waves and the selected threshold and finishes up by finding all the events that apply (using same method as in 'Threshold'). outputs two vectors _E and _A corresponding to detected event times and amplitudes, respectively.

8.1 Inputs and Outputs

Inputs:
_AZ(n)_D : deconvolved wave from ‘New Temporal Profile’ or ‘Old Temporal Profile’
thresh : the threshold for finding events
Outputs:
_AZ(n)_E : nEvents vector of event times
_AZ(n)_A : nEvents vector of amplitudes

9 Cluster

Takes the deconvolved event amplitudes _A and runs a GMM using the EM algorithm. Plots the results of each k-component GMM into a GUI that allows cycling through and selecting the optimal model. This part is still pretty messy - working on fixing the Igor underflow error.

9.1 Inputs and Outputs

Inputs :
_A : the deconvolved event amplitudes
Outputs :
_simMat : matrix of simulated draws from the estimated distributions
_AQ : nEvent vector of estimated quanta for each event
_k(chosen nQuanta)_clustMAP : the most likely quanta for each data point, given k total available quanta/components
_aVals : estimated mixing coefficients
_Mu : estimated quantal means
_S2 : estimated quantal variance

10 Quanta Stuff

10.1 Variables and Nomenclature

nQuanta_i : the set of all events where i quanta are released
maxCycles : the maximum number of cycles of any stimulus block
nStim : the number of different stimulus components
maxQuanta : the maximum quanta of any event

10.2 Inputs

10.3 Ouputs

_QMat : an maxCycle x nStim x maxQuanta matrix of number of events per cycle per stimulus per nQuantal Event
_QMat_Mean : a maxQuanta x nStim matrix of the mean number of events per cycle per stimulus per nQuantal Event
_QMat_Var: a maxQuanta x nStim matrix of the variance of the number of events per cycle per stimulus per nQuantal Event
_QMat_Sum : a maxCycle x nStim matrix of the sum of quanta for each cycle and stimulus
_QTS : the time series of quanta (same dimensions as original recording, but each point is integer valued (zero for no event)

11 Appendix

11.1 EM GMM for Clustering Quanta

We wish to cluster deconvolved peak amplitudes into quanta, which we assume are normally distributed (thus making our question into a gaussian mixture model). In order to do this efficiently, we make use of the EM (Expectation Maximization) algorithm. The wiki page gives a good general description, but I will elaborate on it in the context of our question here.

We will run the algorithm k times (where k is the maximum number of gaussian clusters we are looking for). Each time, we intialize what we think the means could be, as well as what the variances could be (I did this by linearally spacing the means within the range of amplitude values, and setting the ariance as the global variance). We then cycle through two steps, the E step, where we calculate which cluster we think each data point belongs, and the M step, where we maximize our parameters (by recomputing the means and variances) given our cluster occupancies estimated in the E step. We repeat these steps until some convergence is reached, and then we plot all the results.

12 Figures

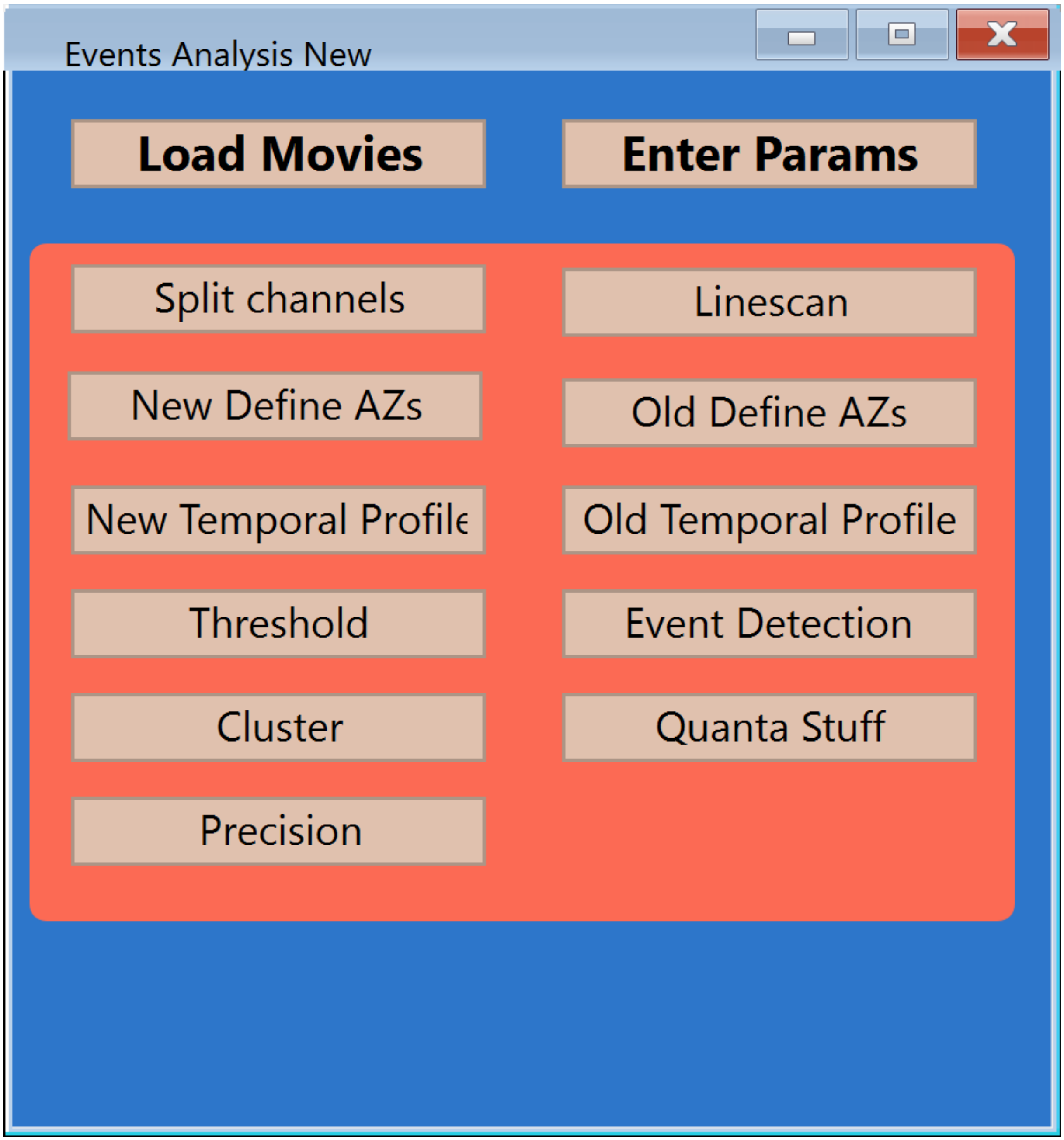

Figure 1: The Glue Sniffer Panel

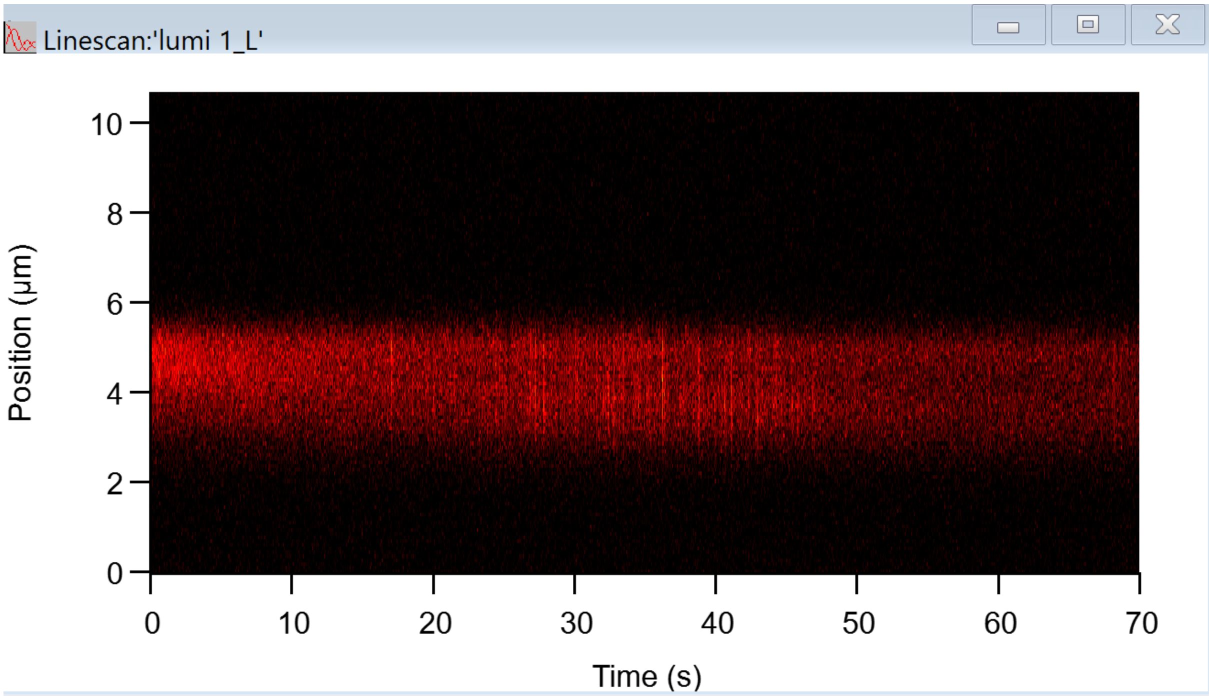

Figure 2: Example Linescan. Output of 'Linescan' Button

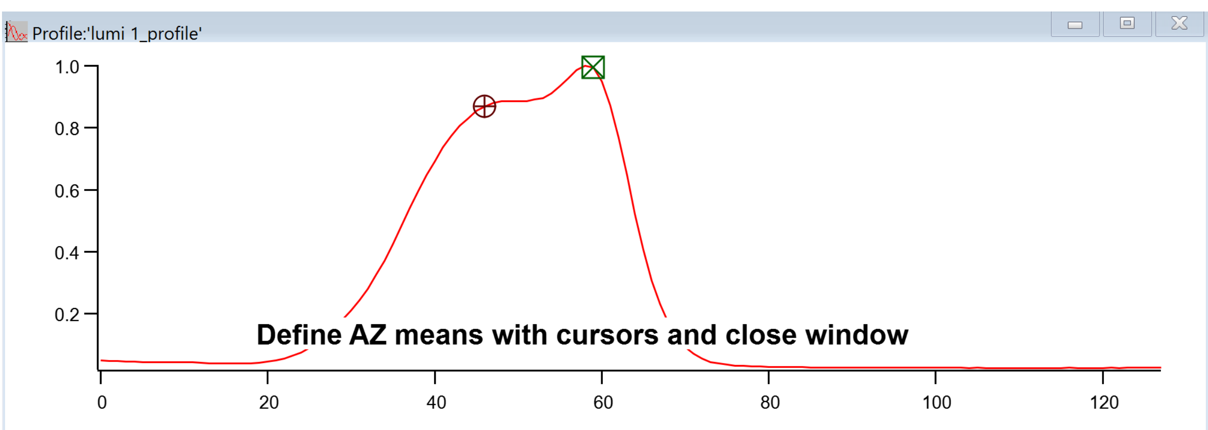

Figure 3: GUI for 'Define New AZs' button

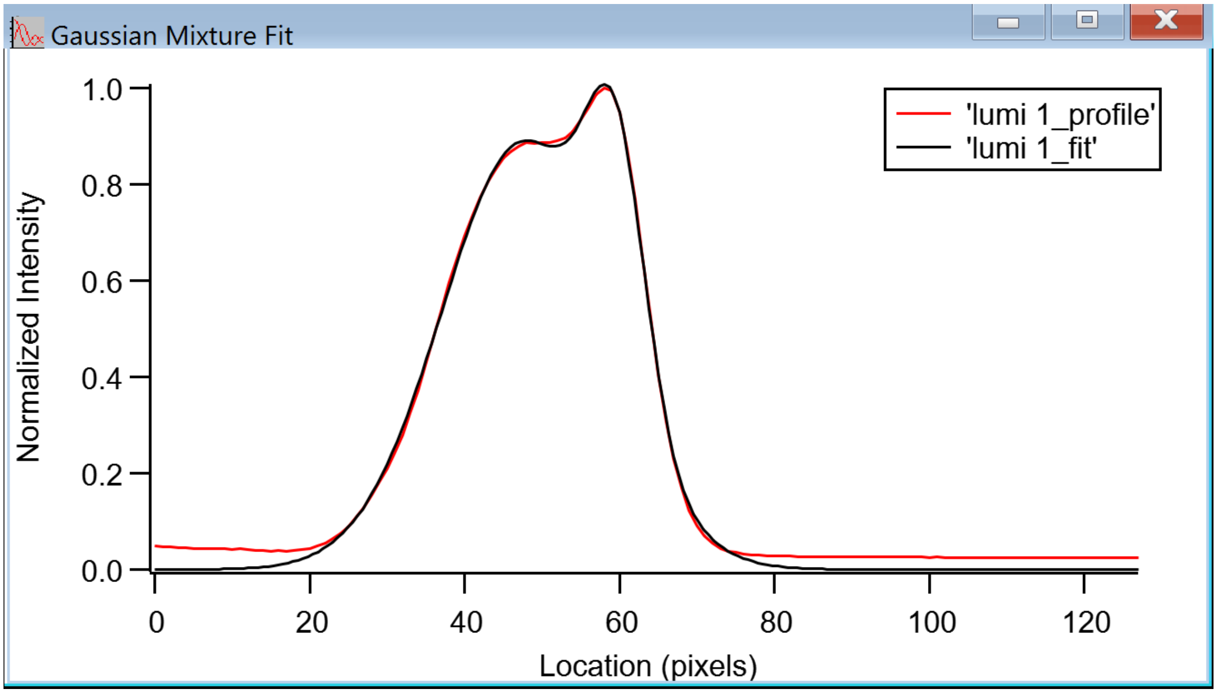

Figure 4: Output of 'New Define AZs' button

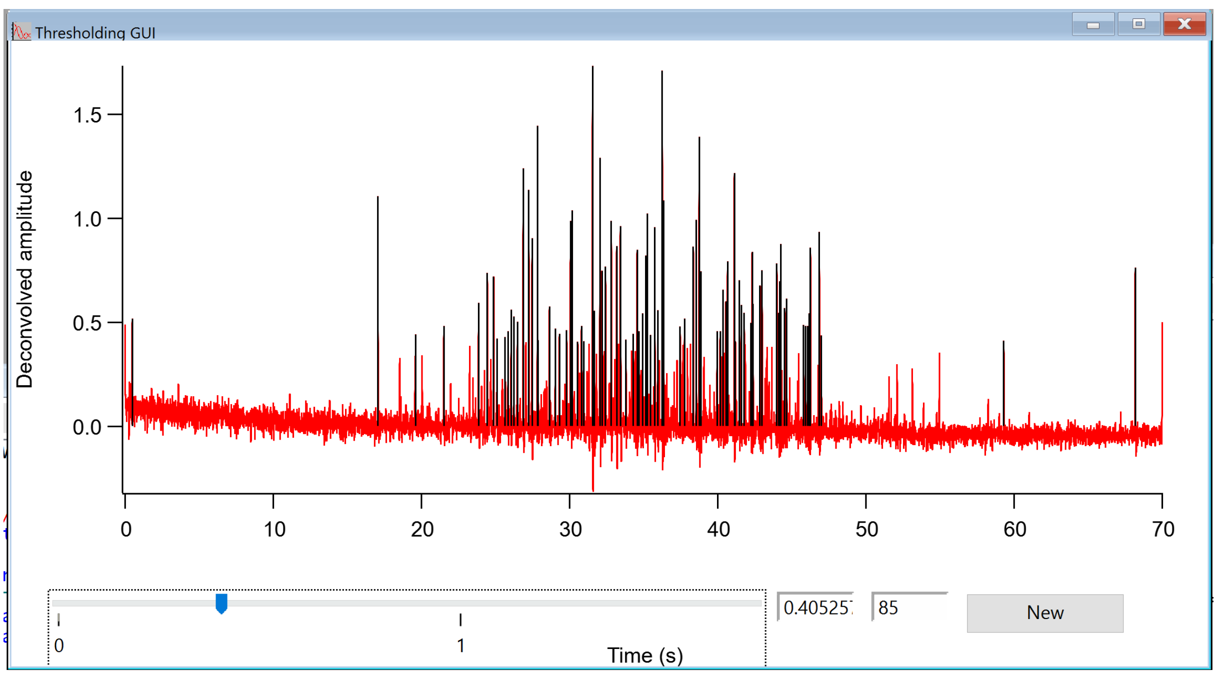

Figure 5: Thresholding GUI

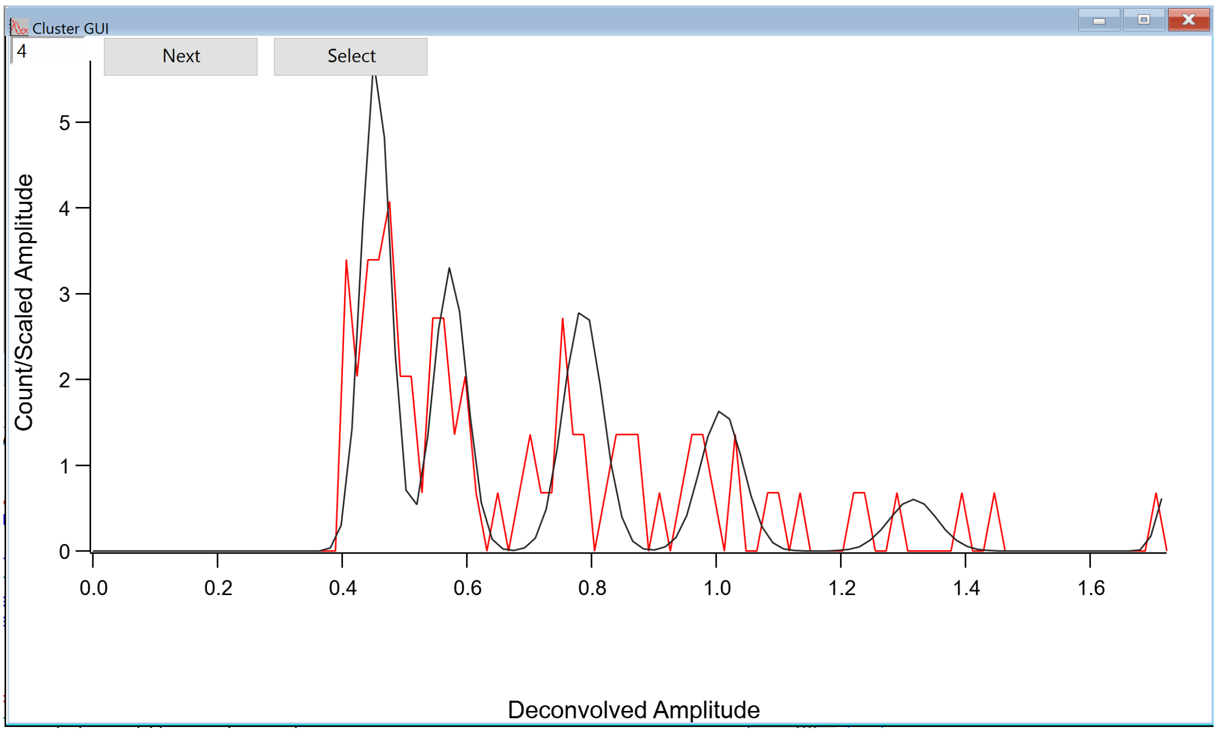

Figure 6: Clustering GUI