**Supplementary Methods**

Overlay thresholding algorithm

SamuROI can use a two-dimensional grey scale morphology image to automatically create a mask around relevant objects in the morphology.

The mask can be visualised from the GUI and has to be validated manually. It is calculated as follows:

1. compute percentile of morphology image and use it as initial threshold value
2. apply Sobel filter (from scikit image) to morphology image and create an elevation map
3. create a marker image where pixels get marked that are close to the threshold
4. run watershed algorithm (from scikit morphology)
5. user inspects result and potentially adapts threshold value

For the specific implementation see the SamuROIData implementation and the scikit image documentation.

Fluorescence renormalization

SamuROI comes with three different ways to renormalize the raw fluorescence data F which depends on space (x,y) and time (t):

1. via standard deviation: in this case the video data is segmented into different blocks, each containing B frames. Then the standard deviation is calculated for every pixel in each block over the frames of that block. Then the block with the minimum standard deviation is selected for each pixel and the baseline fluorescence is calculated as the mean of these blocks resulting in a baseline image F0(x,y)
2. via a linear fit: To account for small bleaching effects a linear fit is calculated for the time series of each pixel. With the resulting slopes mx,y and offsets ox,y, F0 is defined as F0(x,y,t)=mx,yt + ox,y.
3. via the median over individual frames: in this case F0 does not depend on spatial coordinates and is simply defined as F0(t)=median(F(x,y,t)), where the median operates on all pixels of individual frames.

In all cases the calculation of ∆F/F = (F-F0)/F0 from the above definition is implemented via numpy’s array broadcasting functionality.