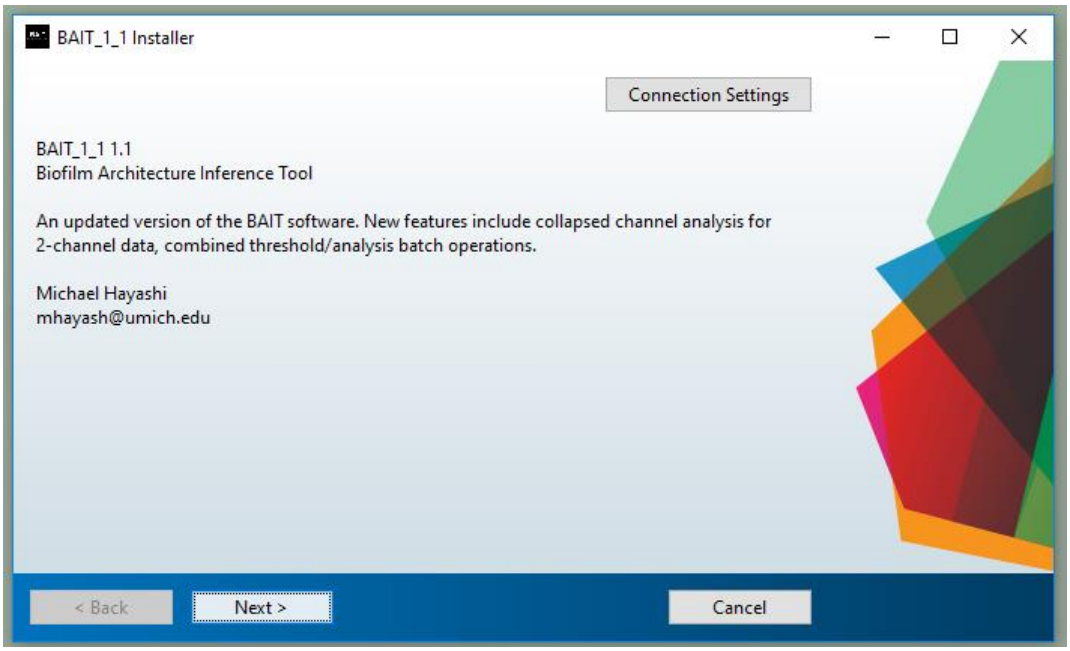
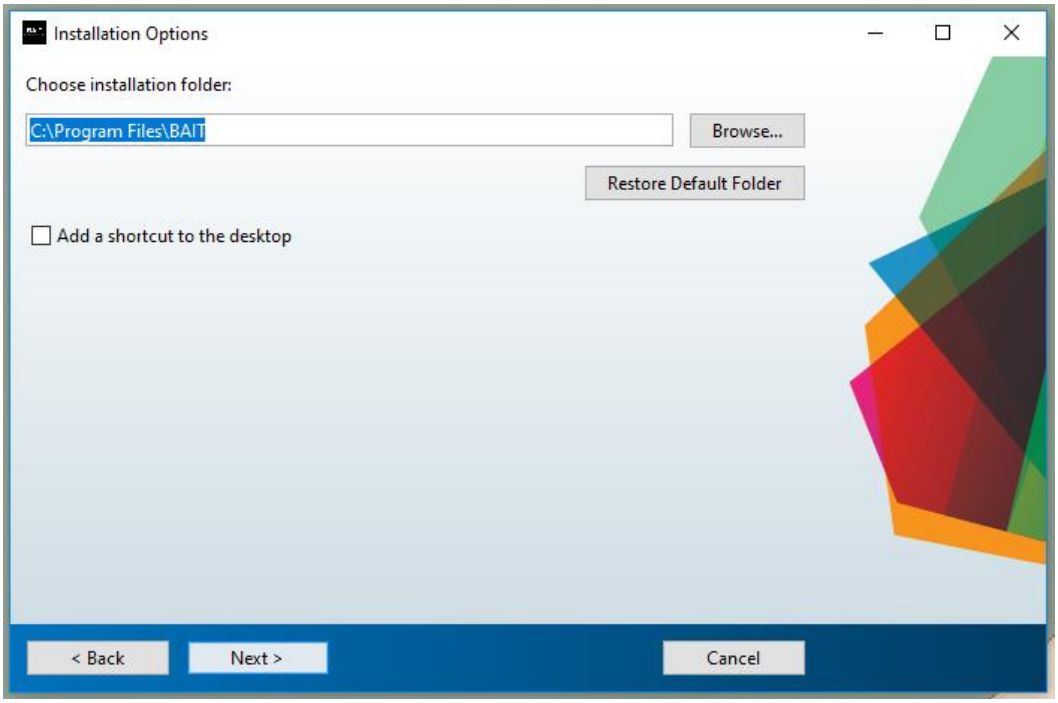
Biofilm Architecture Inference Tool (BAIT) v1.1 and v1.0 Documentation

Installation

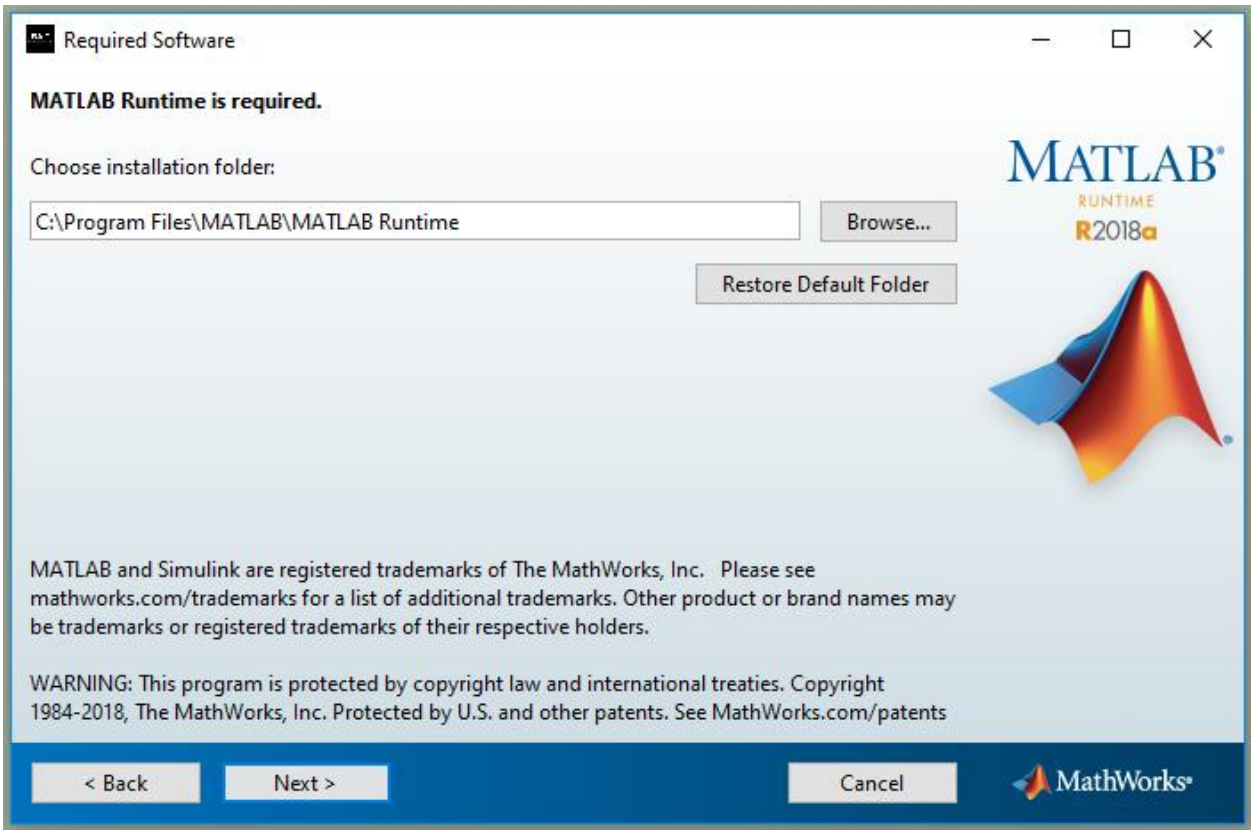
- Download the web installer. BAIT installers are available at https://github.com/epimath/BAIT_software/releases
- Run the installer



- If desired, change the installation directory (recommended for systems with separate OS/Storage partitions)



- If needed, select the location for the MATLAB Runtime



- To continue installation, the MATLAB runtime license terms will need to be accepted

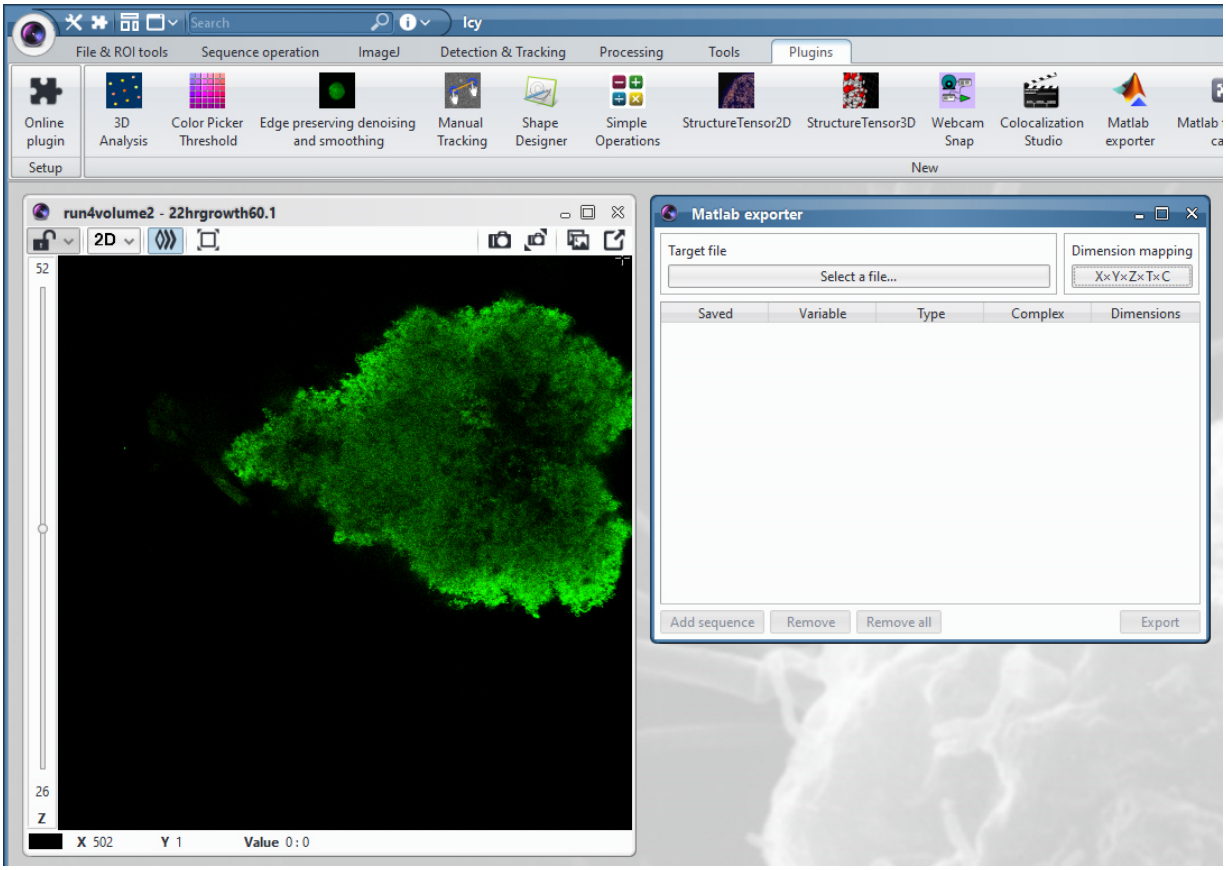
Creating BAIT Input Files

BAIT accepts MATLAB formatted data in *.mat* file format. Specifically, the *.mat* file is a cellular array with the dimensions [X, Y, Z, T, C] where X, Y, Z are Cartesian coordinates, T is time, and C is the corresponding channel. This version of BAIT can be used to analyze single-channel data or to analyze dual-channel LIVE/DEAD staining data. For the dual channel LIVE/DEAD data, T is restricted to 1 and C is limited to 1 (DEAD) or 2 (LIVE) for appropriate viability calculations

Converting image stacks to *.mat* files

A free software that can convert image stacks to acceptable *.mat* files is Icy (1). Icy can be downloaded from: http://icy.bioimageanalysis.org/. After installation, install the 'Matlab exporter' plugin under the 'plugins' tab. After loading your image stack(s) and the Matlab exporter plugin in Icy, your interface will look similar to the image below (**Figure 1**). To begin, click 'Select a file' and navigate to a folder destination where the *.mat* file will be saved. Name your *.mat* file and be sure to add *.mat* to the file name as well as the file type in the drop-down menu. Next, click 'Add sequence' and select the image stack(s) that are loaded in Icy to add to your *.mat* archive. Ensure each image has two channels before adding, 'dimension mapping' is set to 'X x Y x Z x T x C' and click 'Export'.

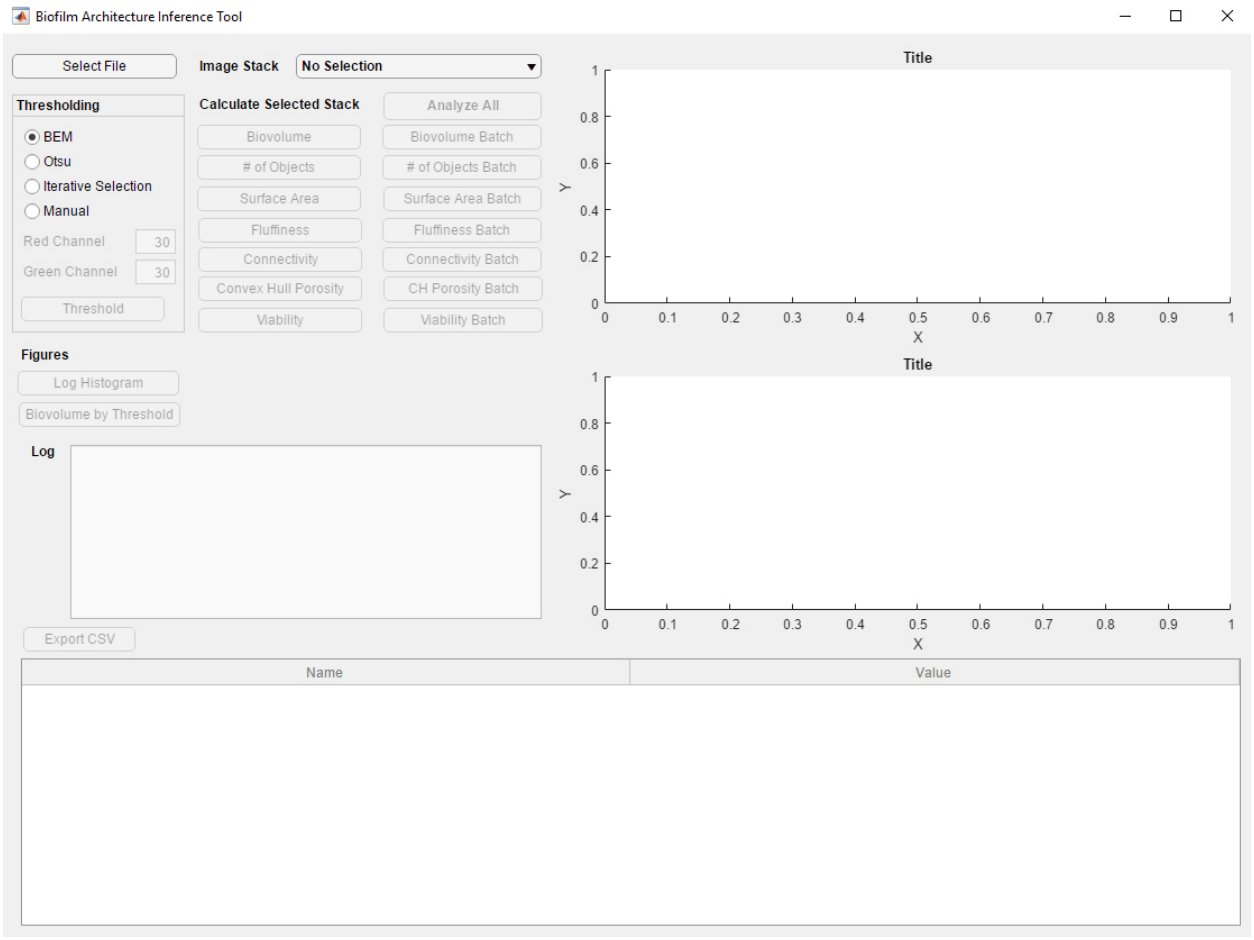
Figure 1



Loading *.mat* files into BAIT

The image below shows BAIT 1.1 graphical user interface (**Figure 2**). To load *.mat* files, click 'Select File' and navigate to the folder containing your *.mat* files. The image(s) loaded can then be selected from the dropdown menu labeled 'Image Stack'.

Figure 2


Biofilm Architecture Inference Tool
Select File
Image Stack
No Selection
Thresholding
BEM
Otsu
Iterative Selection
Manual
Red Channel
30
Green Channel
30
Threshold
Calculate Selected Stack
Biovolume
# of Objects
Surface Area
Fluffiness
Connectivity
Convex Hull Porosity
Viability
Analyze All
Biovolume Batch
# of Objects Batch
Surface Area Batch
Fluffiness Batch
Connectivity Batch
CH Porosity Batch
Viability Batch
Figures
Log Histogram
Biovolume by Threshold
Log
Export CSV
Title
Y
X
Title
Y
X
Name
Value

Thresholding

Thresholding, or image segmentation, is the process whereby signal from a confocal image stack is categorically classified as presence or absence of biological material. Consider a slice of a 3D matrix as shown below:

Table 1a

2	3	4	4
3	2	1	6
1	5	11	8
14	4	10	7
7	10	6	27
12	15	25	10
3	28	15	24
24	8	26	19
27	42	52	11
104	35	11	7

Each cell contains an 8-bit grayscale value from 0-255 indicating intensity of channel signal whereby 0 is black and 255 is the most brilliant signal achievable. A threshold is a cutoff value whereby signal intensity below or equal to the cutoff is categorized as interstitial space and signal intensity above the cutoff is categorized as biological material. For example, a threshold of 4 would result in the binary image below, whereby 1 indicates the presence of a voxel:

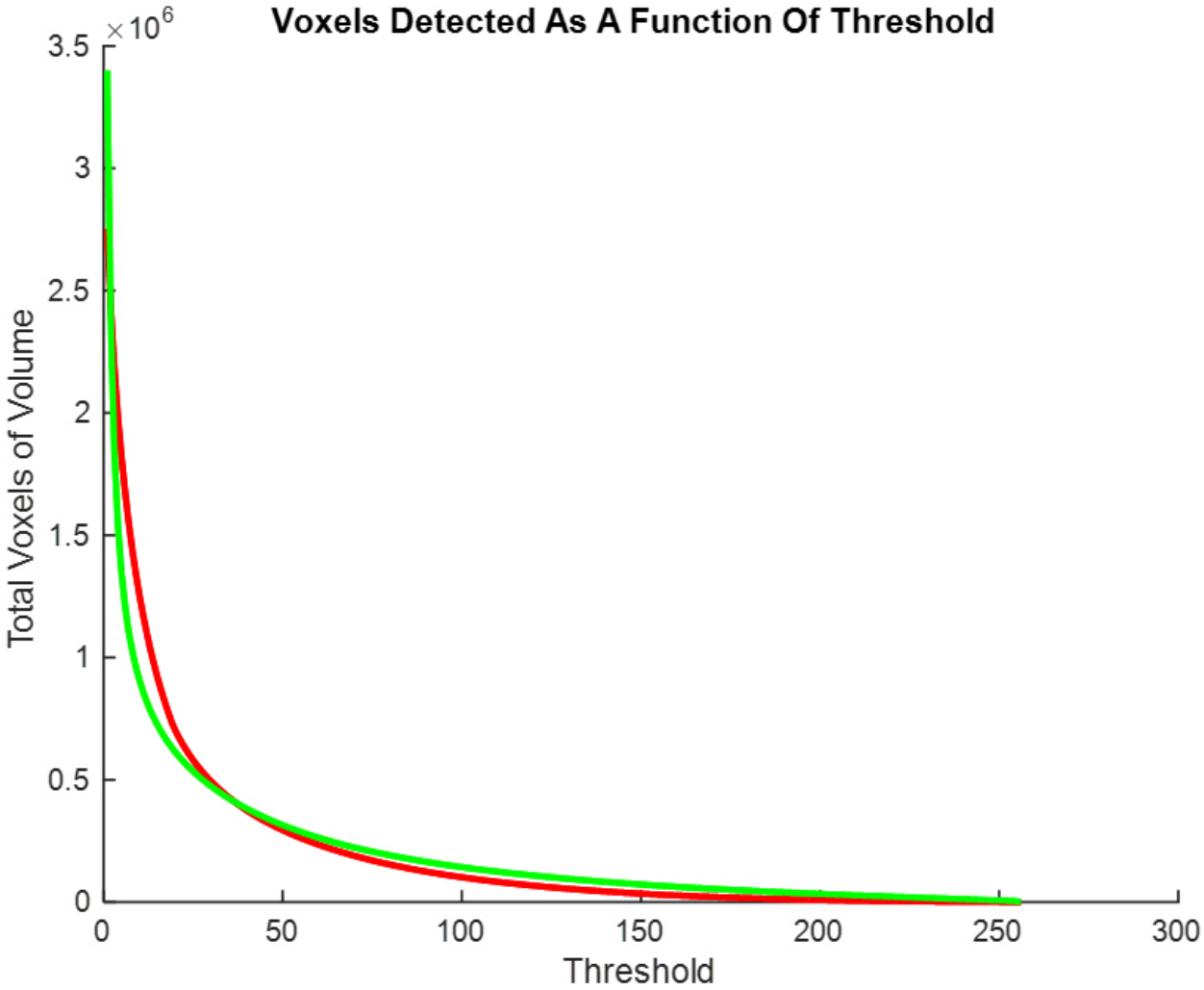
Table 1b

0	0	0	0
0	0	0	1
0	1	1	1
1	0	1	1
1	1	1	1
1	1	1	1
0	1	1	1
1	1	1	1
1	1	1	1
1	1	1	1

Biovolume-Elasticity Method (BEM) Thresholding

The biovolume elasticity method calculates an optimal threshold for fluorescent images by operating on the biovolume by threshold curve. Biovolume is defined as the summation of all '1' voxels in a binary image file after a threshold has been applied. Thus 256 biovolume values are possible for each channel in an 8-bit confocal stack. The biovolume of a confocal stack within a channel decreases as a power-law function of threshold (**Figure 3**).

Figure 3



Total biovolume detected is very sensitive at the lower end of the threshold continuum, indicating noise attributable to spurious background staining. The optimal threshold to detect presence of biofilm material is the portion of this curve where total biovolume becomes relatively inelastic to changes in threshold. We call this threshold the inflection threshold and define it as

the first instance of threshold X where there is less than 10% observed change in slope from threshold X to threshold X+1. For this particular confocal image, the inflection threshold of the green channel is 17 and the inflection threshold of the red channel is 10. The inflection thresholds are used to create a 3-dimensional binary image of dimensions 512x512xZ, whereby Z varies depending on the thickness of the biofilm imaged. For more information regarding the BEM, consult the manuscript by Luo et al (2).

Otsu's Method and Iterative Selection Thresholding

BAIT also has the capability of calculating thresholds using Otsu's method and Iterative Selection. These two algorithms operate on the image histogram, which is the total count of pixels (2D) or voxels (3D) by signal intensity. Otsu's method seeks a threshold that minimizes within class variance. Iterative selection seeks a threshold that maximizes the difference in within class means. In Table 1c, a threshold of 4 would yield the 8-bit matrix below with the class 'biological material' colored green. Within and between class statistics can then be calculated. For more information about these methods please refer to the papers by Otsu (1979), Yang et al., (2001), and Ridler and Calvard (1978) (3-5).

Table 1c

2	3	4	4
3	2	1	6
1	5	11	8
14	4	10	7
7	10	6	27
12	15	25	10
3	28	15	24
24	8	26	19
27	42	52	11
104	35	11	7

Manual Thresholding (v1.1)

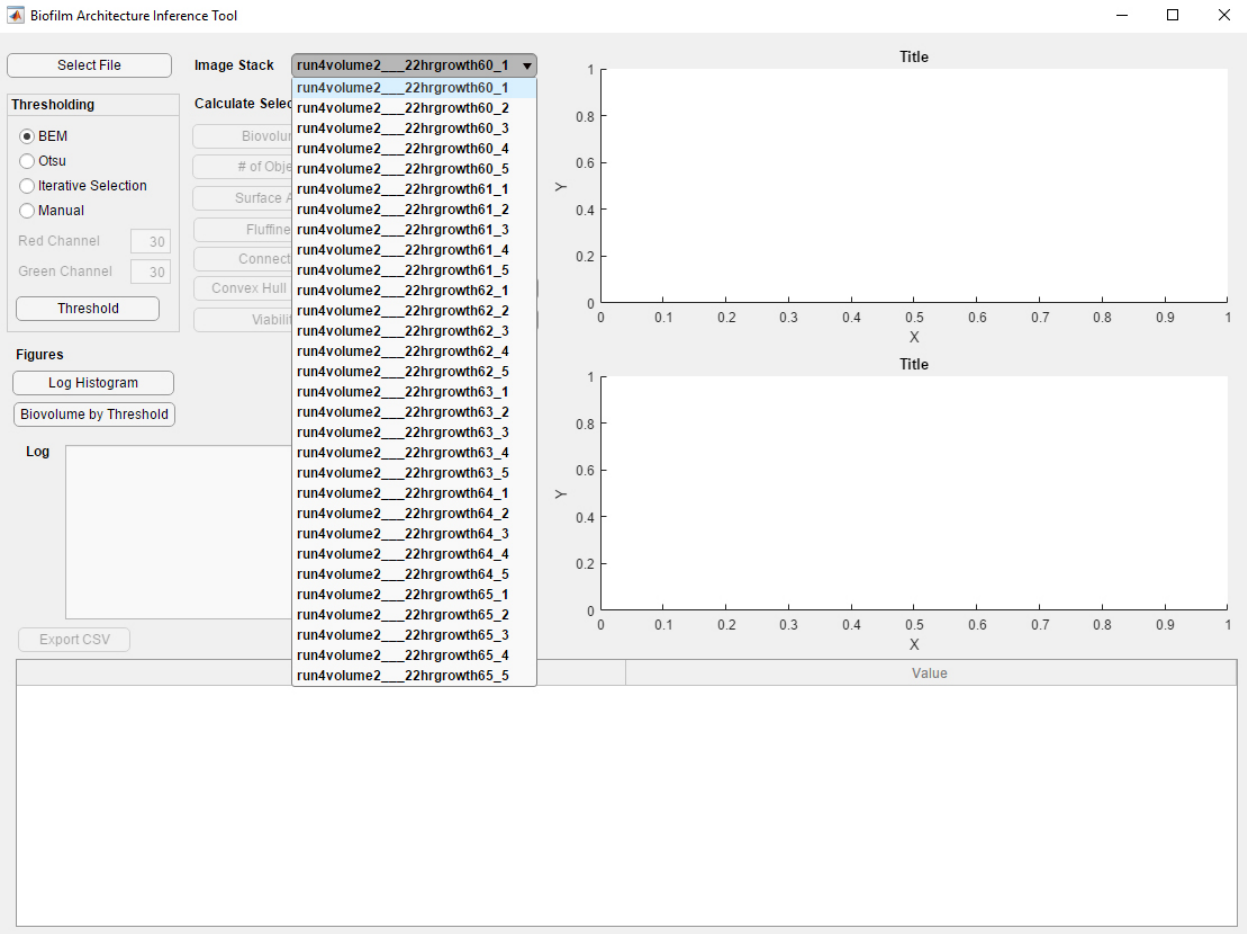
BAIT version 1.1 also enables investigators to manually set thresholds on their image stacks. To manually set a threshold for each channel, enter a value between 0-255 on the indicated 'Red Channel' and 'Green Channel' text boxes (**Figure 2**).

Batch Thresholds and Outcomes in BAIT (v1.1)

BAIT version 1.0 and 1.1 can apply thresholds to individual images or a batch of images. After loading the *.mat* file, the image stacks dropdown menu is available for the user to select individual images from that archive. Once an individual image stack is selected, the 'Threshold' button will be enabled (**Figure 4**). After calculating thresholds of the two channels within an image stack, individual outcomes for that image stack can be calculated. There are three algorithm-driven thresholding methods available in BAIT as well as manual thresholding (available beginning in version 1.1).

An alternative method is batch calculate outcomes based on a thresholding method. To do so, select a thresholding method or manually set thresholds, then click 'Analyze All', or any of the batch outcome buttons (**Figure 2**). The 'Analyze All' button will calculate all outcomes for every image stack in the *.mat* archive. The batch outcome buttons will calculate only the selected outcome for ever image stack in the *.mat* archive (available beginning in version 1.1).

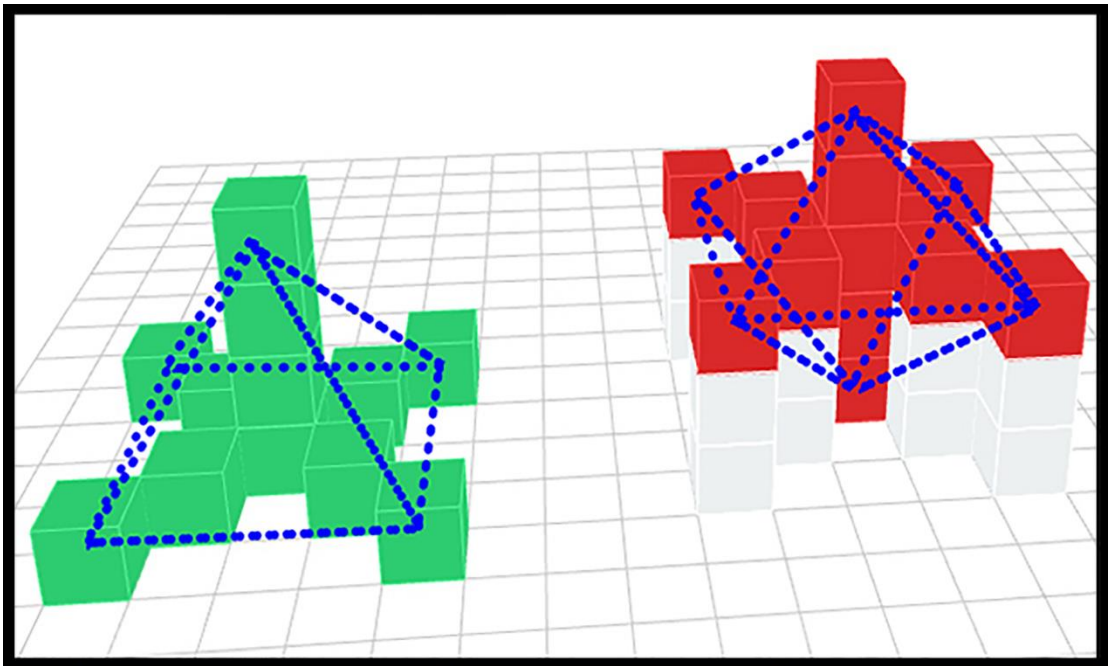
Figure 4



BAIT Outcomes

There are currently six architectural outcomes available in BAIT (6). They are: biovolume, surface area, total number of objects detected, connectivity, fluffiness, and convex hull porosity. These six architectural outcomes can be calculated individually for each channel. BAIT also calculates viability to quantify the amount of dead/damaged cells. Viability requires both the live and dead channels for use. **Figure 5** (from Luo et al. 2019) below will provide a visual representation on how each outcome is calculated in BAIT (6).

Figure 5



Outcome	Green Channel	Red Channel
Biovolume	12 voxels	13 voxels
# of Objects	1 total object	1 total object
Surface Area	66 surfaces	70 surfaces
Fluffiness	5.50 SA/BV	5.38 SA/BV
Connectivity	12 BV/object	13 BV/object
Convex Hull Volume*	16 voxels	21.33 voxels
Convex Hull Porosity	0.25	0.39
Viability	0.48	

Biovolume

Biovolume (and all other outcomes described below) is calculated after thresholds are calculated for each channel. The threshold is applied whereby all 8-bit values within the confocal stack equal and below the value is replaced with 0 and all 8-bit values above the threshold is replaced with 1 to produce a binary image. As with all the outcomes described below, all singleton voxels are filtered out as an additional processing step to increase signal to noise ratio (2). The resulting binary 3D image of the confocal stack is then summed to obtain biovolume. The biovolume of the binary image slice shown in **Table 1b** is 30 pixels.

For 3-dimensional image stacks (**Figure 5**), voxels are the units of the biovolume outcome. It is also calculated by summing the cells that contain a '1' after a threshold has been applied and singleton voxels removed. The green channel biovolume of the image stack represented in **Figure 5** is 12 voxels and the red channel biovolume is 13 voxels.

Surface Area

Surface area is defined as the sum of all biofilm voxel surfaces that are exposed to the environment (bulk-liquid phase and substratum). A singleton voxel floating in 3D space has 6 exposed surfaces. The expected surface area calculations for the two objects featured in **Figure 5** is 66 exposed surfaces for the green object and 70 exposed surfaces for the red object.

Total Number of Objects Detected

The total number of objects detected (also referred to as number of objects or # of objects for brevity) is defined as the sum of all objects completely detached from other objects, obeying the 26-neighborhood connectivity rule. A single voxel in 3-dimensional voxel space has 26 possible neighbors. The measure can be segregated into channels. In **Figure 5**, the total number of objects in the green channel is one and the total number of objects in the red channel is also one. Total number of objects is only an outcome specific to one channel. It does not confer the addition of objects between multiple channels.

Connectivity

Connectivity is a secondary outcome, requiring the measurements of biovolume and total number of objects detected. Connectivity is defined as biovolume over total number of objects (i.e. the average biovolume of all biofilm objects detected). It is a measure approximating biofilm fragmentation. In **Figure 5**, the green channel has a connectivity of 12 voxels per green object. The red channel has a connectivity of 13 voxels per red object.

Fluffiness

Like connectivity, fluffiness is a secondary outcome requiring the measurements of exposed surface area and biovolume. Fluffiness is an estimator of surface area to volume ratio of the biofilm. The measure is defined as the ratio of surface area to biovolume. More specifically, it is the total number of exposed surfaces of the biofilm over the total number of biofilm voxels. The measure is between 0 and 6 since the maximum exposed surfaces of a biofilm voxel is 6. In **Figure 5**, the fluffiness of the green channel is 5.50 and for the red channel is 5.38

Convex Hull Porosity

Convex hull porosity is an estimator of biofilm object irregularity and is defined as the percentage of the convex hull volume of an object that is interstitial space. First, the convex hull volume of an object is calculated by connecting all vertices of the object (dotted blue lines in **Figure 5**). The convex hull volume represents the general presence of the biofilm object. Its value cannot be smaller than the actual biovolume of the object. The volume of the object that is interstitial space can be determined by taking the difference of the convex hull volume and the biovolume of the object. High convex hull porosity values can indicate several biofilm properties. Firstly, the biofilm object could possess many deep crevices. Secondly, the biofilm could contain holes. Thirdly, the biofilm objects could be hollow. In **Figure 5**, the convex hull porosity of the green object is 0.25 whereas the convex hull porosity of the red object is 0.39.

Unweighted Viability

Unweighted viability (also referred to as viability for brevity) outcome is only available for dual channel studies where channel 1 (red) represents compromised cells and channel 2 (green) represents viable cells. The outcome is calculated as the proportion of green voxels to all voxels (green and red voxels). In **Figure 5**, the viability of the image is calculated to be 0.48.

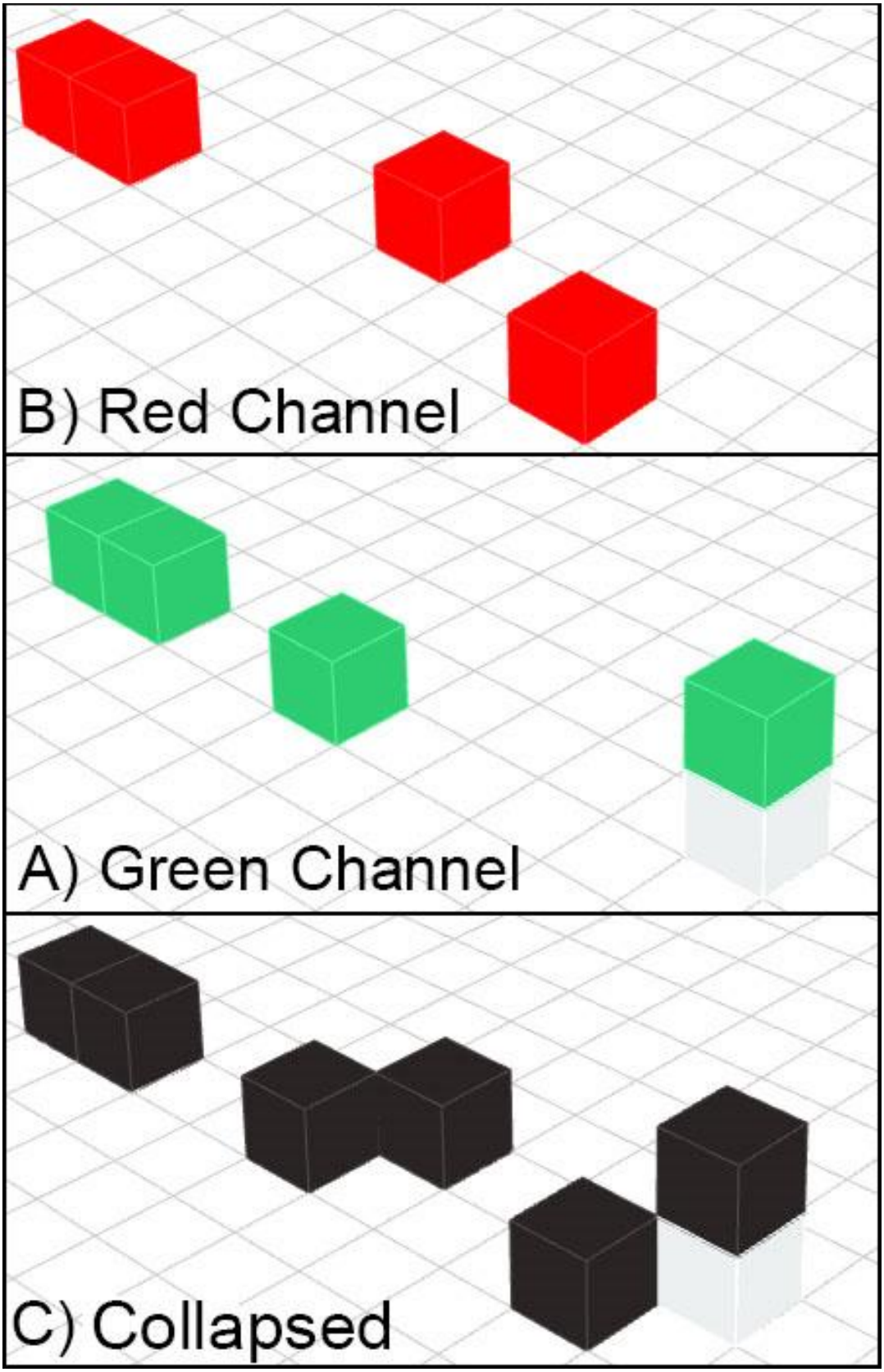
Collapsed Channel Outcomes (v1.1)

For dual channel studies, the two channels can be collapsed (combined) into a singular channel (available beginning BAIT v. 1.1) where the biovolumes of both channels are collapsed into one. This reduces the 12 architectural outcomes possible in dual-channel analyses (biovolume, surface area, total number of objects detected, connectivity, fluffiness, and convex hull porosity) down to 6 for collapsed channel analyses.

For each channel, thresholds are applied, followed by singleton voxel removal to create two binary image files representing the two channels. These two binary image files are then collapsed. **Figure 6** illustrates how collapsing channels work in BAIT after the binary image files are created in each channel. Biovolume in voxel-space that exists in either the red channel or green channel is copied into a third channel (collapsed channel). If a certain voxel-space contains both red and green signal, then the voxel-space of the collapsed channel will contain one voxel, not two. Architectural outcome calculations can then be performed on this third channel.

Collapsed channel biovolume, along with red and green channel biovolumes, can be used to deduce the total number of voxels co-localized within the same voxel-space (BAIT-calculated output is expected to be available beginning BAIT 1.2). A colocalization percentage can then be expressed in terms of the green, red, and collapsed channels. For example, the green biovolume of **Figure 6** is 4 voxels and the red biovolume is also 4 voxels. The collapsed channel biovolume is 6 voxels, indicating that 2 voxels are colocalized within the same voxel-space (the left most object in **Figure 6**). We can then say 50% of the green voxels are green alone and the other 50% of the green voxels are co-localized with red. Similarly we can say 50% of the red voxels are red alone and the other 50% of the red voxels are co-localized with green. Lastly, we can say 33% of all voxels are co-localized.

Figure 6



Tips & Tricks

- Before using BAIT, make sure the scale of the desktop is set to 100% in the Display settings Window. BAIT could load off screen if your desktop zoom is set to more than 100%.
- Be aware loading time for BAIT can take up to a minute.
- Depending upon the imaging system and file types, the use of BAIT will require additional software. For example, to convert files to Matlab executables for analysis in BAIT, the software program ICY (icy.bioimageanalysis.org) can be used.
- Ensure that image stack names are saved beginning with a letter. If image stack names begin with a number or underscore, then import of *.mat* file into BAIT will fail. Icy can change image stack names before export into *.mat* files.
- For viability measurements, make sure the correct channels are assigned. The red (dead) channel should be the 1st channel (X,Y,Z,1,**1**) and the green (live) channel should be the 2nd channel (X,Y,Z,1,**2**). If your image stacks default to green being in the first channel and red in the second, check your confocal configuration and alter protocol as needed. Our confocal protocol lists 'Scan 1' as the propidium iodide channel and 'Scan 2' as the Syto-9 channel.
- Check that image stacks are in 8-bit as these are best suited for thresholding with BEM. BAIT can operate on higher-bit images, but manual thresholding is recommended.
- For analysis of multiple images within an experiment, ensure that all image stacks are of the same bit and resolution.

Recommended System Specs:
Windows 10, Intel Pentium computer with at least two cores and 8GB of ram.

References

1. de Chaumont F*, et al.* (2012) Icy: an open bioimage informatics platform for extended reproducible research. *Nature methods* 9(7):690-696.

2. Luo TL*, et al.* (2018) A Sensitive Thresholding Method for Confocal Laser Scanning Microscope Image Stacks of Microbial Biofilms. *Scientific reports* 8(1):13013.

3. Otsu N (1979) Threshold Selection Method from Gray-Level Histograms. *Ieee T Syst Man Cyb* 9(1):62-66.

4. Yang X, Beyenal H, Harkin G, & Lewandowski Z (2001) Evaluation of biofilm image thresholding methods. *Water research* 35(5):1149-1158.

5. Ridler TW & Calvard S (1978) Picture Thresholding Using an Iterative Selection Method. *Ieee T Syst Man Cyb* 8(8):630-632.

6. Luo et al. 2019. Introducing BAIT (Biofilm Architecture Inference Tool): a software program to evaluate the architecture of oral multi-species biofilms. Microbiology, in press.