

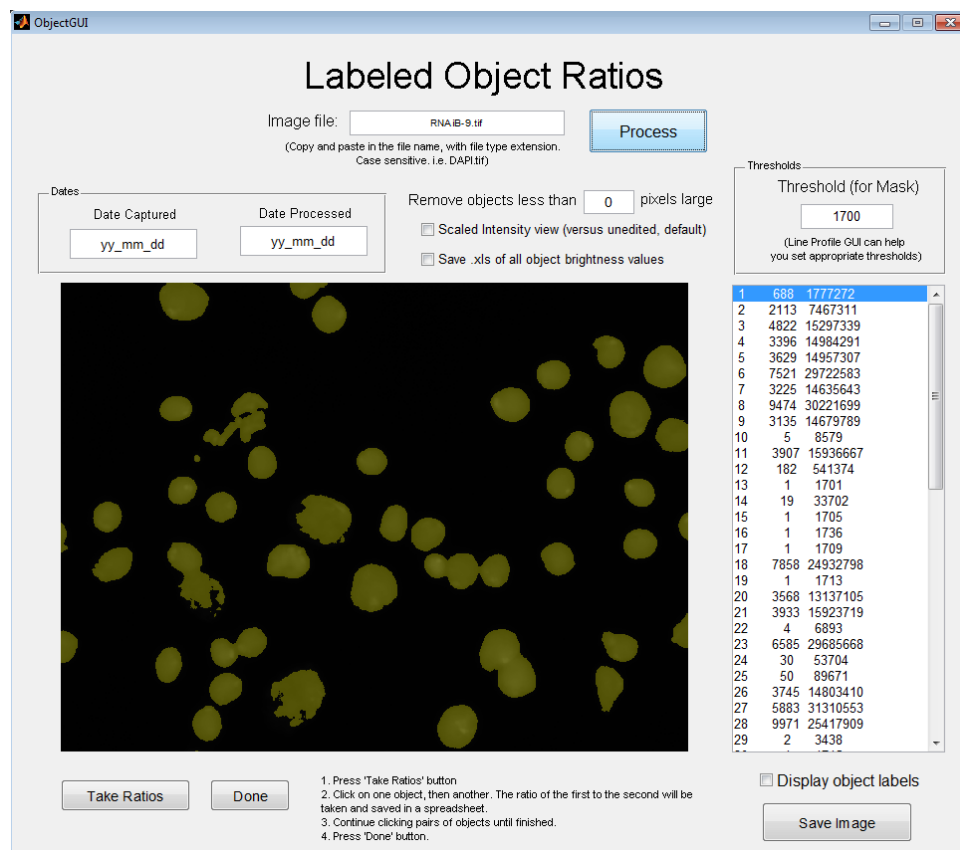
# 1 Instructions for Object Differentiation Matlab GUI

Liana Engie

15 July 2011

These are the instructions for using Liana Engie's Fluorescent Object Differentiation Graphical User Interface (GUI) on Matlab.

The program is designed to take a fluorescence image and quantify the intensity of brightness of the images. Background negation is done by masking out everything that is not the object of interest, finding the area of each object, and summing the brightness of individual pixels.



#### Input:

- A greyscale `.tif` images

#### Output:

- A display of the image, with the mask applied, and a list of the objects, their areas, and the brightness.
- (optional) Saved `.png` of the image with the mask applied (and label, if desired),
- An `.xls` spreadsheet with the intensity over area ratios of pairs of user selected objects, and, if desired, the area and intensity of every object created from the threshold.

### **1.1 What the code does**

Here I'll probably reference a paper once it's published. If you're really interested in the details, you should probably read my thesis. But, basically:

This program examines single images, treats them as greyscale, and differentiates areas of interest, or objects, using a threshold that blocks out every pixel underneath that value. The program shows where the objects are using a yellow false color overlay, and labels each object with a number, starting from left top to right bottom. The objects' area, in pixels, and fluorescence signal intensity are listed to the right of the image. The objects' labels can be toggled on and off, and will be displayed in blue at the center of the object, on the image. The image can also be saved, and a spreadsheet of the objects and their information can be generated if desired.

The user can choose to select pairs of objects, and these objects' fluorescent intensity over area (brightness density, in a sense) will be ratioed, and these numbers will be saved in a spreadsheet.

#### **1.1.1 Output outside of the GUI**

After you process the images using the GUI, the program will automatically generate an `.xls` file that saves the raw calculated data and ratios. The name of the file is generated using the Date Captured: `'image_name.Processed.xls'`.

If you do not check the ‘Save spreadsheet’ checkbox or take any ratios, no spreadsheet will be produced.

## 1.2 Opening Matlab and the GUI

This code was written in Matlab v.2007a. When opened, you should see the Command Window - probably the largest window - and possibly also the Current Directory, Command History, and maybe the Workspace. To use the GUI you only need to work with the Command Window and Current Directory.

The image that you want to analyze should be in your Current Directory, along with the files `ObjectGUI.m` and `ObjectGUI.fig`. They just have to be in the same folder on your computer, and navigate the Current Directory to that folder. You can move files around directly within the Current Directory.

To open the GUI, either type

```
>> ObjectGUI
```

into the command window (case sensitive), or drag `ObjectGUI.m` from the Current Directory into the Command Window. The blank GUI should open:

## 1.3 Before Processing

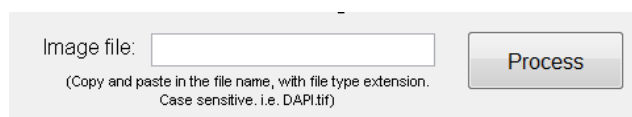
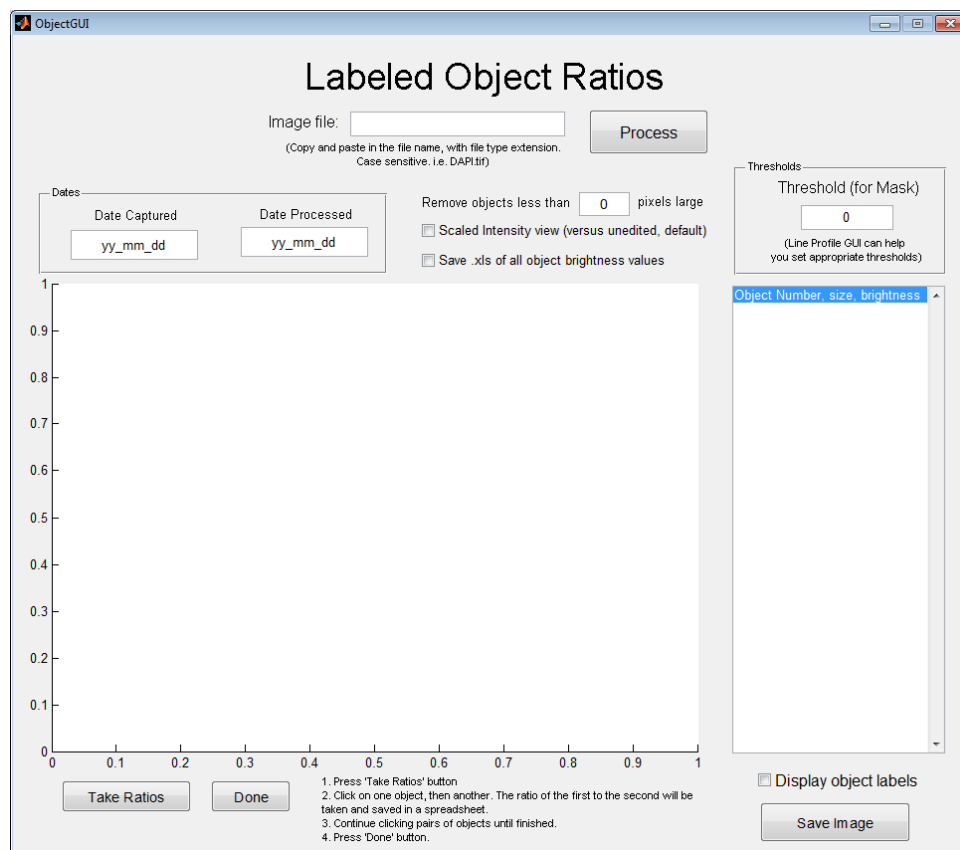
Before the image is processed, the variables need to be set.

### 1.3.1 Inputting an image

Copy and paste the name of the file from the Current Directory to the blank at the top of the interface, or manually type in the name, including the file extension.

### 1.3.2 Setting Dates

Input the date the image was captured and the date processed (both will be recorded in the spreadsheet, but are not critical to anything). If you do not enter anything it will remain ‘yy\_dd.mm’.



### 1.3.3 Setting the Threshold

This number will determine how wide the mask's coverage of the objects of interest is (see 'What the code does', above).

(If you change the threshold values while the analysis is running, or afterwards, the images will not be affected. The changed threshold will only come into effect after you press 'Process' again.)

The **Line Profile GUI** can help you determine what the thresholds should be.

### 1.3.4 Removing objects under a certain size

Depending on the image and the threshold, occasionally a larger object will be lined with several objects of just a few pixels. Depending on what you are using the program for, you may not want these objects cluttering the list box, and can remove objects under a certain pixel area, say, 20 or 50 pixels, by putting this value into the blank.

(If you change this value while the analysis is running, or afterwards, the list will not be affected. The items under that value's area will only be removed after you press 'Process' again.)

### 1.3.5 ‘Scaled view’ of image

Clicking on this checkbox will display the image with scaled range of brightness intensity. Usually, the image displays with highest possible value (depends on the image’s bit depth) being absolute white, and the lowest possible value (0) being black. Scaling takes the individual image’s range of values and scales the black and white colormap to it, thus making it easier to see images on the ends of the spectrum of absolute values, and differentiate features of an image.

*Scaling the brightness does not change the numeric brightness of the image - just the visual aspect.*

### 1.3.6 Saving the values in a spreadsheet

If you would like to save all of the objects and their information outside of the user interface, checking this box will have the program generate a `.xls` file with the objects and their area and fluorescent intensity listed. A file is automatically generated when ratios of pairs of objects are taken. If you have chosen to save the raw numbers, the ratios will be saved on another sheet within the same file. If you have not chosen to save the raw information, a new file will be made.

The file name is generated from the image name: `name_Processed.xls`

If you are processing the same image several times with different thresholds or other variables and want to save previously generated spreadsheets, you will need to resave them under different names, else they will be written over.

### 1.3.7 Processing and force quitting

Once everything is set, press the ‘Process’ button to the right of the Image Name input blank to start the analysis of the image. This may take a few seconds.

The false color mask (yellow) will be overlayed on the image, and a list of the objects, from top left to right bottom, will be displayed in the listbox on the right. The list will display the objects’ pixel area and fluorescent signal intensity.

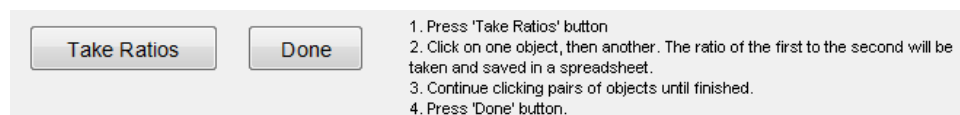
If, while the code is processing, you want to stop it for any reason (realized that there is an extra or missing image messing up the reading frame, or that the threshold(s) need to be changed, etc) go back to the main Matlab window, click on the Command Window (the main window), and press `Ctrl + C`.

## 1.4 After Processing

### 1.4.1 Toggling the object labels images

### 1.4.2 Saving the image

### 1.4.3 Taking Ratios



If you have chosen to save all of the raw numbers in a spreadsheet, the ratios will be saved in a separate sheet in that file. If not, a new 'image\_name.Processed.xls' file will be generated. Each line will show the label for the first and second objects and their ratioed "brightness density".

### 1.4.4 Recalculating data

You can recalculate the same data by just pressing the 'Process' button again. If you're changing the threshold and choosing the option of saving the values in a spreadsheet, the next dataset will save over the old one. If you want to keep the last dataset, you need to resave the .xls file under another name, as it will be rewritten (if the Date Captured remains the same).

If you want to calculate another image, copy and past that into the blank and press 'Process' again. The image must be in the Current Directory (the folder that is open) If not, copy and paste it into there or close the GUI, go to the folder of interest within the Current Directory window, making sure `ObjectGUI.m` and `ObjectGUI.fig` are both in the folder, and reopen `ObjectGUI.m`.

## 1.5 Errors

Errors may crop up, such as if you select "Done" after an odd number of selections for pair ratio-ing. Closing and reopening the program should always fix the problem.