

# Merging fluorescence images

Hernan G. Garcia

March 30, 2009

## 1 Introduction

In this tutorial we will show how to merge the images of a fluorescent sample taken using different channels into one composite color image. When quantifying images it is usually better to work with the independent channels. However, it is useful to be able to display all channels at once. Combining different channels can also be useful when looking for colocalization of different fluorophores.

## 2 Loading and setting up the images

We start by loading the images corresponding to the three fluorescent channels. You'll be most likely working with a DAPI, FITC, and TRITC snapshots of the same cell. We'll assign DAPI to the blue channel, FITC to the green channel, and TRITC to the red channel. This assignment is related to the actual wavelengths of the emission of the fluorophores.

```
ImR=imread('brain_60x_700ms_TRITC_wheels.tif');  
ImG=imread('brain_60x_400ms_FITC_wheels.tif');  
ImB=imread('brain_60x_200ms_DAPI_wheels.tif');
```

Remember that these images have a bit depth of 16 bits and that the range of each pixel goes between 0 and 65535. First, we convert this range to the  $[0, 1]$  range that Matlab uses by applying the function `mat2gray`

```
ImR=mat2gray(ImR);  
ImG=mat2gray(ImG);  
ImB=mat2gray(ImB);
```

Note that applying this function rescaled each one of the images based on their minimum and maximum pixel values. After applying this function

we've lost the quantitative information stored in the absolute value of each pixel in each channel. We'll assume that we're making an image for displaying purposes (in your notebook, for example) so that this rescaling is not of importance.

### 3 Combining the different images

In order to combine them we'll use the Matlab function `cat`. This creates a structure that can be directly interpreted by `imshow` as a color image.

```
ImRGB=cat(3,ImR,ImG,ImB);
```

Where the "3" in the command is an option to set the dimension of this concatenated structure. Finally, we can show the whole image using `imshow(ImRGB)`.