COLOCALIZR - USER MANUAL

ALLAN SAUVAT
allan.sauvat@gustaveroussy.com

MARION LEDUC
marion.leduc@gustaveroussy.com

February 15, 2019

1 Introduction

ColocalizR is an R-based image-analysis application developed for the quantification of co-localization in high-throughput. ColocalizR is aimed at assessing co-distribution at the organelle level and thus can be used with widefield microscopes.

NB: ColocalizR has been optimized for the use of the nomenclature and SQL database architecture of images acquired with the Molecular Devices MetaXpress software, but can be used on any other set of images.

2 Requirements

ColocalizR has been developed and tested on Windows, Linux and MacOS X operating on 64-bit systems. We recommend a computer with a minimum of 8 GB of RAM and equipped with a quad-core (or higher) central processing unit (CPU).

During installation an internet connection is required for downloading necessary R packages.

3 Installation

3.1 Software installation

3.1.1 Windows

To use this app in Windows, R, RStudio, Java and Strawberry Perl have to be installed.

3.1.2 MacOS

R and RStudio can be installed in the same way as under Windows configuration. Java is not required in MacOS but Xcode is necessary and can be installed either by download via the AppStore or by command line via the terminal. The easiest way in MacOS 10.9 or later is to copy the following into the Terminal app:

```
xcode-select --install
```

3.1.3 Linux

In Linux, program files can be installed in the same way as in the two previous configurations or by using the commands below to correctly install and configure R, RStudio, Java and Perl.

Install dependencies

```
sudo apt install libcurl4-openssl-dev libssl-dev unixodbc unixodbc-dev libtiff-dev fftw-dev
    fftw3 fftw3-dev libv8-3.14-dev perl
```

Install R

```
# Ubuntu
grep -q -F "deb http://cran.rstudio.com/bin/linux/ubuntu *UbuntuVersion*-cran*Rversion*/"
    /etc/apt/sources.list || sudo echo "deb http://cran.rstudio.com/bin/linux/ubuntu
    *UbuntuVersion*-cran*Rversion*/" >> /etc/apt/sources.list
sudo apt-key adv --keyserver keyserver.ubuntu.com --recv-keys E084DAB9
sudo apt update
sudo apt install r-base
```

```
# Debian
grep -q -F "deb http://cran.rstudio.com/bin/linux/debian *DebianVersion*-cran*Rversion*/"
    /etc/apt/sources.list || sudo su -c "echo 'deb http://cran.rstudio.com/bin/linux/debian
    *DebianVersion*-cran*Rversion*/' /etc/apt/sources.list"
sudo apt install dirmngr
```

```
sudo apt-key adv --keyserver keys.gnupg.net --recv-key
    'E19F5F87128899B192B1A2C2AD5F960A256A04AF'
sudo apt update
sudo apt install r-base-
```

Install RStudio

```
wget https://download1.rstudio.org/rstudio-*RStudioversion*-amd64.deb
sudo dpkg -i rstudio-1.1.456-amd64.deb
sudo apt install -f # install missing dependencies
```

Install Java

```
sudo apt install -y default-jdk
sudo R CMD javareconf
```

3.2 R packages

Some packages have then to be installed before launching the application, by running the lines below in the R or RStudio console:

```
install.packages(pkgs=c('rJava','devtools'), repos = "http://cloud.r-project.org")

# For R versions < 3.5
source("http://bioconductor.org/biocLite.R")
biocLite(pkgs=c('EBImage','flowCore'), ask=F)

# For R versions >= 3.5
if (!requireNamespace("BiocManager", quietly = TRUE))
   install.packages("BiocManager")
BiocManager::install(c("EBImage","flowCore"), version = "3.8")

devtools::install_github(c("kroemerlab/MetaxpR", "kroemerlab/MorphR",
    "kroemerlab/ColocalizR"))
```

4 File type and nomenclature

Images are automatically annotated and stored as described below if they were acquired using the Molecular Devices MetaXpress software. In that case, no further formatting is necessary to perform the analysis with ColocalizR.

Images coming from other sources must be in tag image file format (TIFF), and have to be annotated according the nomenclature detailed as follows:

One plate, one timepoint. In that case the images must be stored in a folder, named corresponding to an integer defining *Plate ID*. Each image should be named according to its acquisition parameters; it must contain the experiment name, the wellID (from A01 to P24), the siteID (when several images are acquired in the same well), and the acquisition channel (1,2, or 3): *ExpName_Well_si_wi*.

```
Plate ID
├── MyExp_A01_s1_w1.tif
├── MyExp_A01_s1_w2.tif
├── MyExp_A01_s1_w3.tif
├── MyExp_A02_s1_w1.tif
└── ...
```

One plate, several timepoints. In that case, a subfolder must be created for each timepoint of the experiment: *TimePoint_i*

```
Plate ID
├── TimePoint_1
│   ├── MyExp_A01_s1_w1.tif
│   ├── MyExp_A01_s1_w2.tif
│   └── MyExp_A01_s1_w3.tif
└── TimePoint_2
    └── ...
```

Several plates, one timepoint. In that case, a root folder should be used, inside of which a subfolder needs to be created for each plate.

```
Plate Root
├── Plate_1
│   ├── MyExp_A01_s1_w1.tif
│   ├── MyExp_A01_s1_w2.tif
│   └── MyExp_A01_s1_w3.tif
└── Plate_2
    ├── MyExp_A01_s1_w1.tif
    ├── MyExp_A01_s1_w2.tif
    └── MyExp_A01_s1_w3.tif
```

Several plates, several timepoints. The application was not optimized for that purpose. Nevertheless, applying all of the above guidelines will allow to format the images for analysis by ColocalizR.

5 Analysis workflow

5.1 Launching the app

After installation, the application can be launched by the following commands (in an R or RStudio console):

```
require(ColocalizR)
Launcher()
```

This will open a browser window with the main panel, as displayed in figure 1.

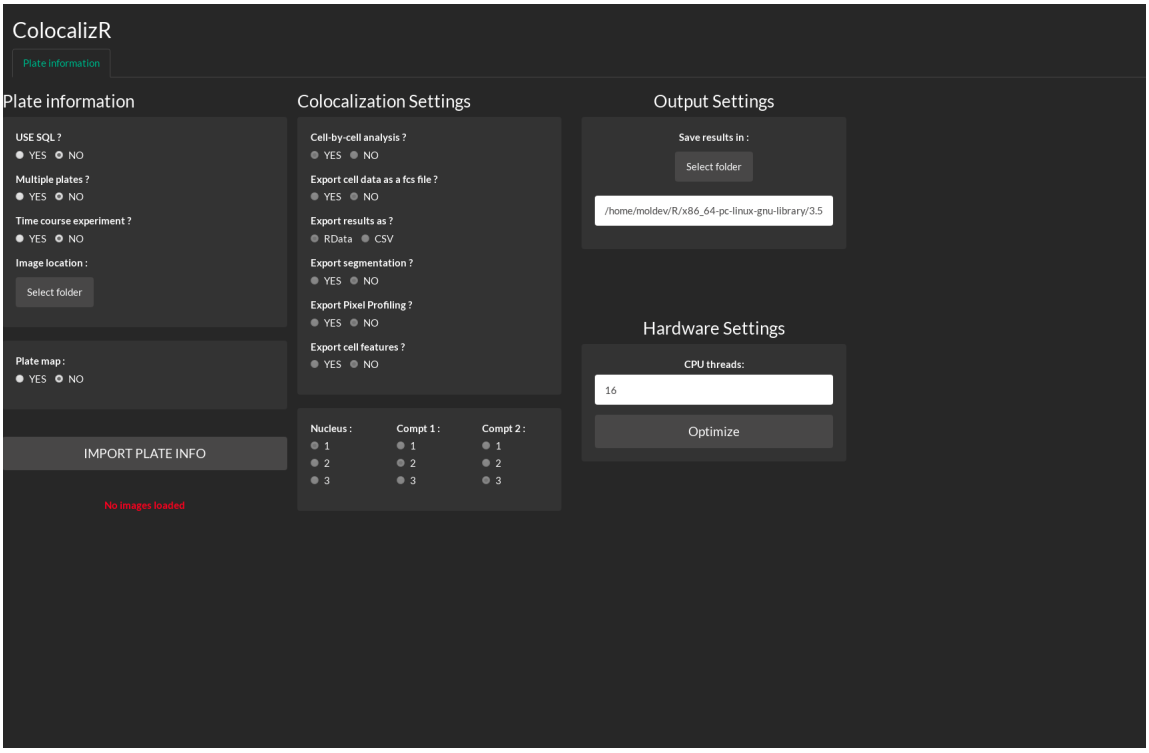

Figure 1: Main page of ColocalizR GUI.

5.2 Main tab: selecting images

Please refer to section 4 for image annotation.

Plate selection. Images can be loaded either by using a SQL connection (this option requires a SQL database architecture), or by indicating their location on the local computer. In the case of SQL connection and if it exists several databases, the one where image informations are stored has to be selected in the **Database** menu. In the case of a local storage, a **Plate selection** window will appear when the **Select folder** button is clicked to choose the correct path to images. Then, **Multiple plates?** and **TimeCourse experiment?** options have to be defined according to experiment parameters. The selection of plates for analysis has to be confirmed by **IMPORT PLATE INFO**. Following, the information will appear in the **Test Settings** menu as shown in figure 2. For navigating through the selected images, it is advisable to import a plate map (**Select platemap**), which has to be in comma-delimited csv format. It should be named with its corresponding Plate ID.

Colocalization settings. First, cell-based or image-based analysis has to be chosen. In the first case, ColocalizR returns one set of coefficients per cell, which can be used for example for clustering subpopulations. This however requires more computing resources and increases the time for analysis. In the second case, the amount of data decreases (as compared to cell-based) and consequently the speed of analysis increases. However, a much higher rate of false negative and positive data has to be expected. Our recommendation is to perform the analysis on a cell basis. For subsequent analysis the generated data can be exported as flow cytometry standard (fcs) files, which will allow using any flow-cytometry analysis software for graphical display. Data will also be exported as *.csv or *.Rdata (R environment) files, for analysis by other means. In addition to colocalization data and in the case of a cell-by-cell analysis, cell features (such as nuclear and cell area, fluorescence intensities in both channels) can also be calculated and exported. Besides, the segmentation as a TIFF file (one site per well), and the pixel profiling, i.e. scatterplot of pixel intensities distribution within the image, can be exported as well.

Image channels. Colocalization quantification requires obviously at least two channels. For performing cell-based segmentation, a supplemental nuclear stain is mandatory. Channels corresponding to either ***"Nucleus"***, ***"Compartment 1"*** or ***"Compartment 2"*** have to be defined as indicated in image name (w1, w2 or w3, see 4). If no nuclear stain is present, it has to be defined as channel 3.

Output settings. The folder where the results will be exported can be chosen by clicking on **Select folder** in the **Save results in** option.

Hardware settings. This window allows to decide how many cores will be dedicated to the analysis. **Optimize** button will calculate the optimal number of cores to be used according to the available RAM on the host machine.

Test settings. Once the plate information has been correctly set and imported, **Test Settings** window appears, where images used to test and adjust segmentation parameters can be chosen (the plate, the time, the well and the site has to be defined).

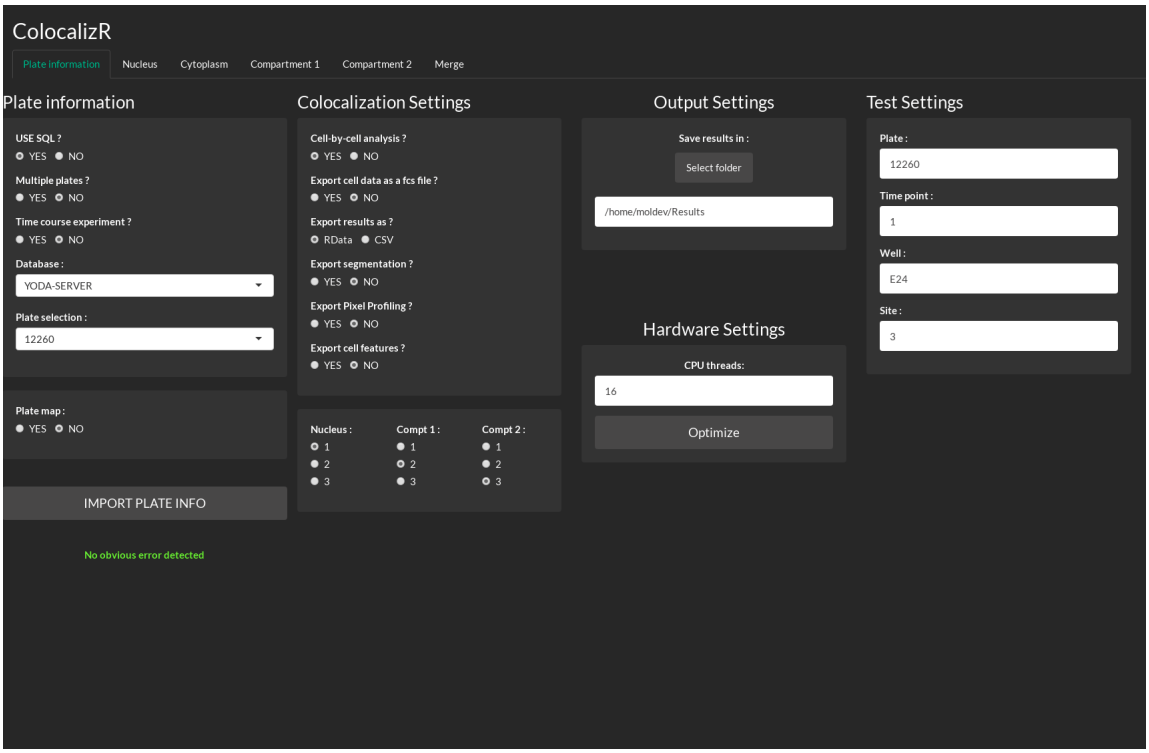

Figure 2: Main page after plate selection.

5.3 Adjusting segmentation settings

5.3.1 Before starting

Once images have been loaded and colocalization settings have been defined in the previous step, the settings tabs (**"Nucleus"**, **"Cytoplasm"**, **"Compartment1"**, **"Compartement2"** and **"Merge"**) display. If not, image location, and/or settings must be reviewed in the **"Plate information"** tab (an error message indicates incomplete or incorrect settings).

5.3.2 "Nucleus" tab

Exclusion of nucleus from mask. If no nuclear staining is present, it is recommended to leave **Remove nucleus from mask** on ***YES***. Otherwise, it is possible to exclude the nuclear region from the colocalization assay which would be applicable in conditions where the signals that are analyzed for colocalization are exclusively cytoplasmic.

Window size. The segmentation of the nuclear compartment is based on an adaptive segmentation method. The principle of this method is the use of a moving squared window to apply a mean filter to the image, and to compare the original image with its filtered version modulated by an **offset** (defined by the user):

$$Mask = Image > (Filtered\ Image + offset)$$

The **Window size** parameter defines the half width and height of this window. If the size is set to 50, the image will be filtered by a 101x101 pixel window. This parameter must be superior to 0 and inferior to the size of the image.

Note: To ameliorate image segmentation, the algorithm uses several window sizes based on the selected size. If window size is 50, thresholding is performed with 51x51, 75x75, 101x101, 125x125, 151x151 pixel windows.

Offset. Here the offset, which ranges from -1 to 1 (with -1 being the least and 1 being the most stringent), has to be defined for segmenting the nucleus.

Segmentation method. Two alternative segmentation methods are available for the nuclear compartment: ***Fast*** and ***Robust***. The ***Fast*** method corresponds to a propagation method based on the Voronoi labelling algorithm, whereas ***Robust*** corresponds to a watershed algorithm performed on the distance map of the analyzed image. Both of these methods result in the separation and labelling of each nucleus. In most cases, the ***Fast*** method will provide satisfying results; but, if cells are too confluent or with a heterogeneous staining, the ***Robust*** method could give better results, yet might increase analysis time.

Image denoising. In the case of an extremely noisy image leading to a poor segmentation (small artifacts detected as nuclei, or holes in nuclei), an additional step of morphological filtering conducted on binary images can be added to the pipeline, by selecting ***YES*** for **Denoise image?**, and choosing a size for structuring element (it must be an odd number, and a structuring element of small size is often more efficient) in **Denoising filter size**. This step consists in a geodesic reconstruction by dilation, then a geodesic reconstruction by erosion, both using a disc structuring element.

Extrema smoothing. This is a parameter allowing an adjustment of the image histogram, removing the lowest and the highest values (extrema), thus, facilitating segmentation. The selected parameter defines the number of greyscales that will be removed (decreasing values corresponds to an increase in the number of greyscale that will be removed (which might influence the calculation of the correlation coefficients): an **Extrema smoothing** of 2 removes 10^{-2} % of the lowest and the highest values from the histogram).

Adjust image. The **Auto-adjust** parameter determines the best range of intensities to visualize the image. To change this range for better visibility, ***NO*** must be selected for **Auto-adjust** and the **Adjust image** slider must be used to adjust the signal range.

Different combinations can be tested by clicking **TEST SETTINGS** and observing the segmentation results.

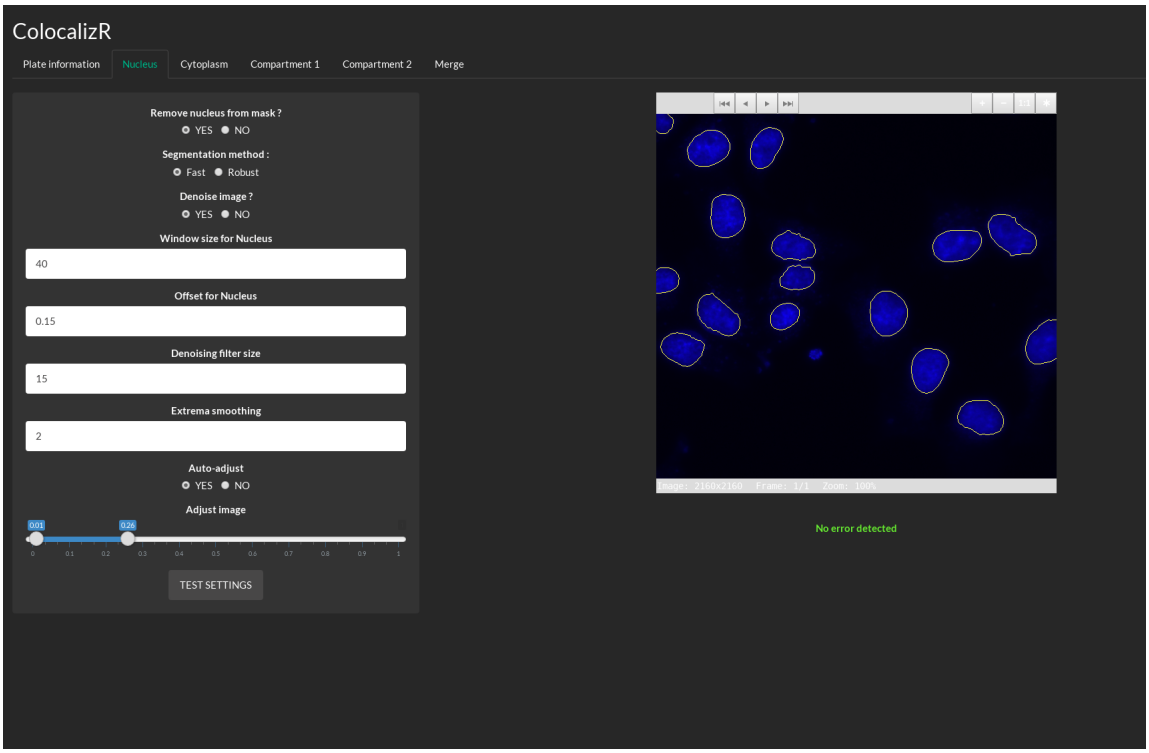

Figure 3: "Nucleus" settings tab.

5.3.3 "Cytoplasm" tab

Channel for cytoplasm segmentation. The cytoplasm can be segmented using the ***Compartment 1***, ***Compartment 2***, or ***Both*** images.

Cytoplasm segmentation method. Cytoplasm segmentation can be performed by an automated method, an adaptive method, or a combination of both. For each of these methods, certain parameters can be adjusted in order to obtain an optimal segmentation of the cytoplasm.

Adjustment parameter. The automated method uses the Otsu's algorithm to automatically determine a threshold for each image. The threshold can be improved by using an adjustment of the stringency of segmentation. This adjustment parameter (**Adjustment for cytoplasm segmentation**) ranges between 0 and 2 (the higher, the more stringent).

Offset and Window size. For the adaptive method, the window size and offset can be adjusted, in the same way as described previously in the **"Nucleus"** section).

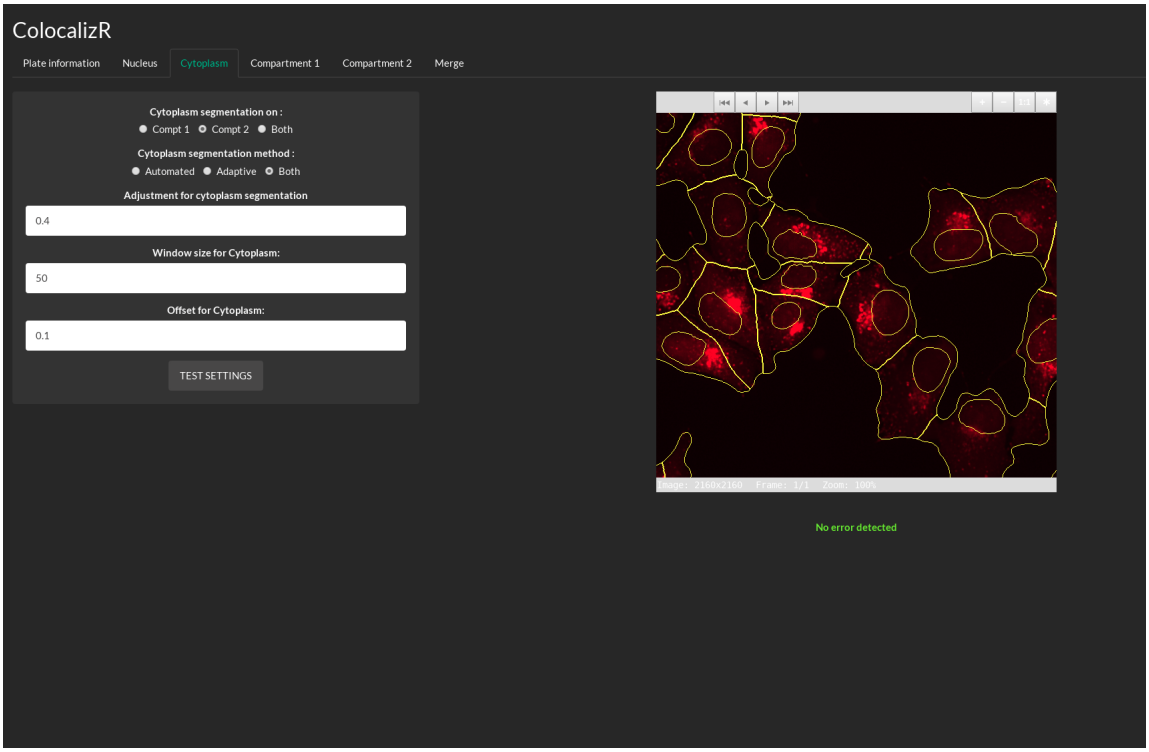

Figure 4: "Cytoplasm" settings tab.

5.3.4 "Compartment 1 and 2" tabs

Automated segmentation. Here the choice between automated and manual segmentation has to be made. We recommend leaving this option set to ***YES***, to avoid any bias due to manual thresholding. Still, the offset can be set manually for refining the channel mask (but the window size is automatically fixed depending on the top-hat filter size). This can be useful when working with surface overlap coefficient rather than Pearson coefficient.

Top-hat filter size. The top-hat filter is a morphological operation used to enhance high-contrast regions, and thus, highlights high-intensity details. The **top-hat filter size** defines the size of the structuring element to be used to perform this operation.

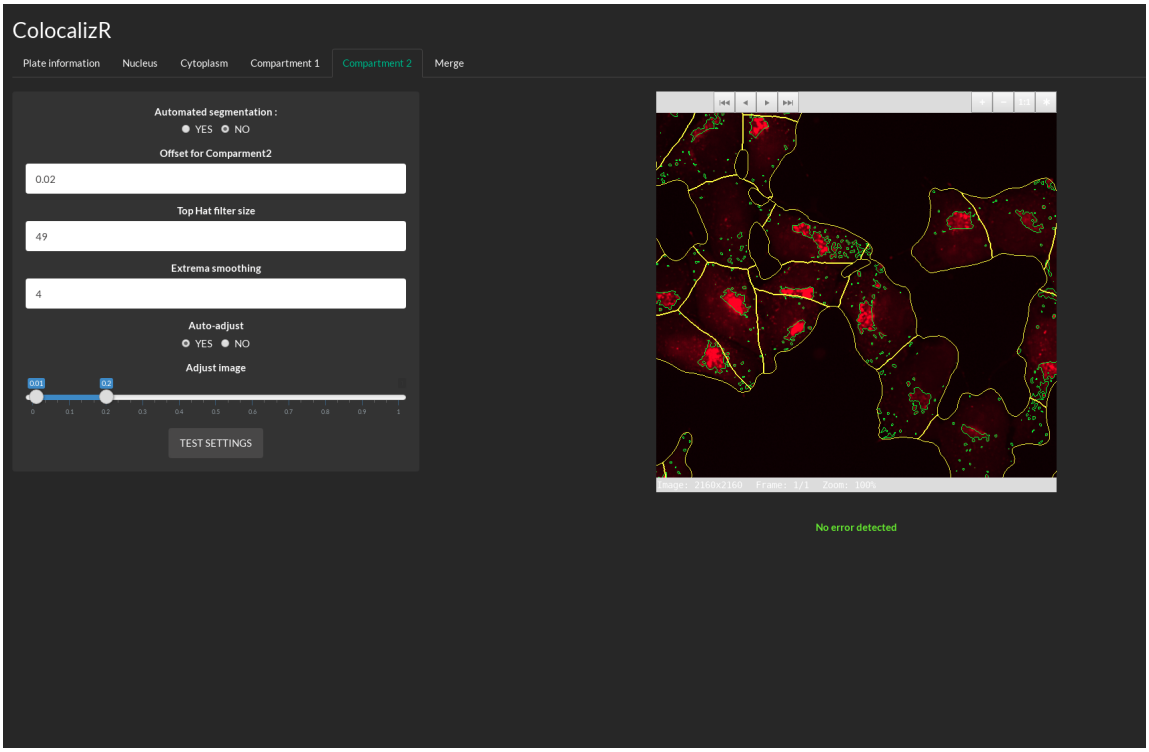

Figure 5: "Compartment 2" settings tab.

5.3.5 "Merge" tab

Here the channels will be displayed as an RGB image, with the segmentation overlay. PCC and SOC calculated on the whole image are displayed, indicating the reliability of the selected settings, that can be refined if necessary.

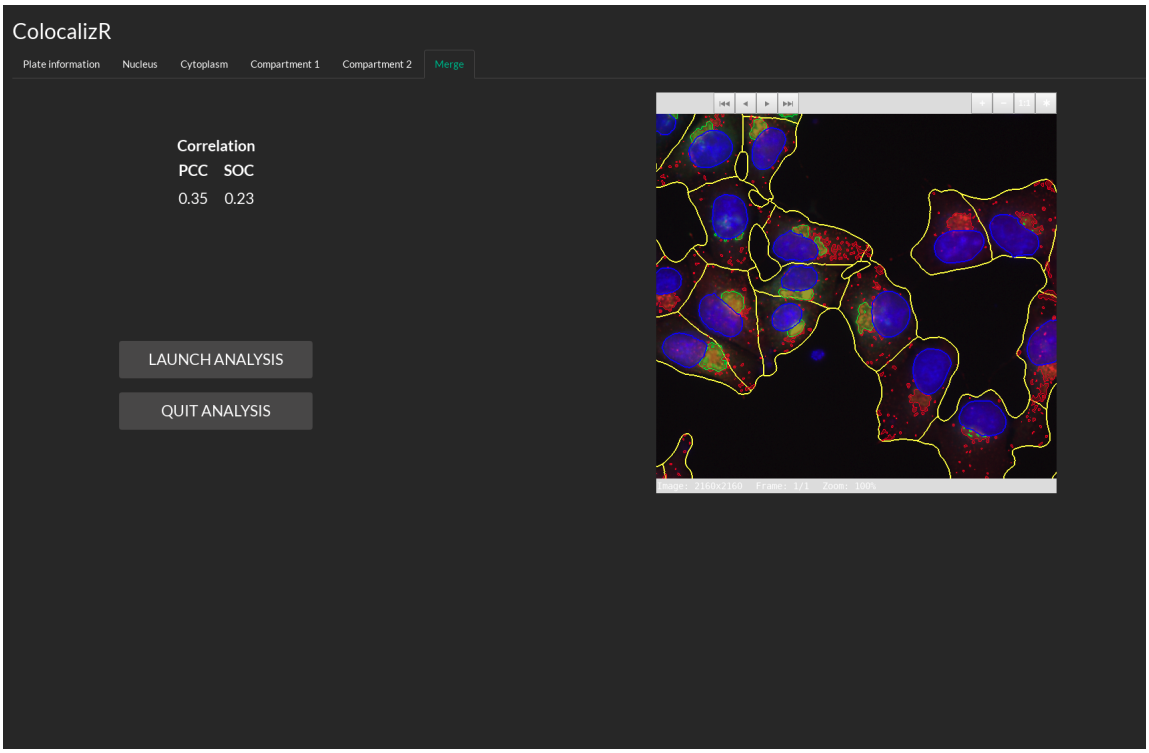

Figure 6: "Merge" settings tab.

5.4 Analyzing Results

5.4.1 Colocalization coefficients

Colocalization can be expressed by different means. Here is a quick summary of the correlation coefficients calculated by ColocalizR.

Pearson Correlation Coefficient. Pearson Correlation Coefficient (PCC) is the most robust and widely-used coefficient to quantify correlation. It is weakly sensitive to signal segmentation. It ranges from -1 (perfect anti-correlation) to 1 (perfect correlation), its formula is described as:

$$PCC = \frac{\sum (R_i - R_{av}).\sum (G_i - G_{av})}{\sqrt{\sum (R_i - R_{av})^2.\sum (G_i - G_{av})^2}}$$

Intensity Correlation Quotient. Intensity Correlation Quotient (ICQ) is derived from PCC. It ranges from -0.5 to 0.5, its formula is described as:

$$ICQ = \frac{\sum (R_i > R_{av}) = \sum (G_i > G_{av})}{N_{pix}} - 0.5$$

Manders Overlap Coefficient. Manders Overlap Coefficient (MOC) differs from PCC in the way that it's taking absolute fluorescence intensities, and not their departure from the mean. It ranges from 0 to 1, its formula is described as:

$$MOC = \frac{\sum (R_i).\sum (G_i)}{\sqrt{\sum (R_i)^2.\sum (G_i)^2}}$$

where R_i is the intensity of the first stained compartment (in red) in individual pixels and R_{Av} the arithmetic mean, whereas G_i and G_{Av} are the corresponding intensities for the second labeled compartment (in green) in the same pixels.

Surface Overlap Coefficient. Surface Overlap Coefficient (SOC) is different from the coefficients described above. Indeed, it does not take into account the pixel intensities, but only their co-occurrence at the same position. By its nature, this method is very sensitive to segmentation. It ranges from 0 to 1, and it can be calculated as:

$$SOC = \frac{\sum P_{G,i} \cap P_{R,i}}{\sum P_{G,i} \cup P_{R,i}}$$

where $P_{G,i}$ and $P_{R,i}$ correspond to the Boolean values indicating the absence (0) or the presence (1) of signal in a single pixel, in green and red channels respectively. $P_{G,i} \cap P_{R,i}$ equals 0 when there is no co-occurrence at a given pixel position (and 1 in the opposite case), and $P_{G,i} \cup P_{R,i}$ equals 0 in the absence of any signal at the same pixel position and 1 if at least one compartment is detected. The fractional SOC for the green and red fluorescence (SOCG and SOCR, respectively) is also calculated by replacing $P_{G,i} \cup P_{R,i}$ with $P_{G,i}$ or $P_{R,i}$.