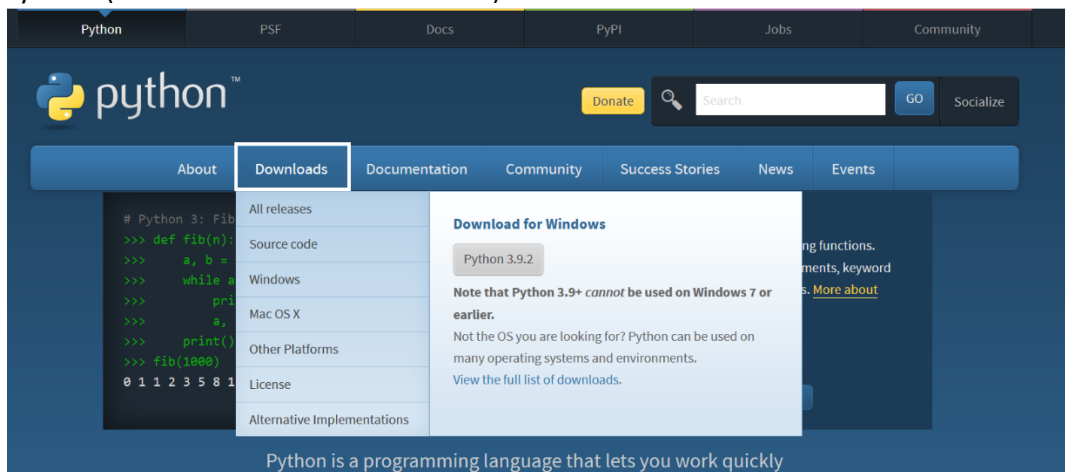


Pre-requisites

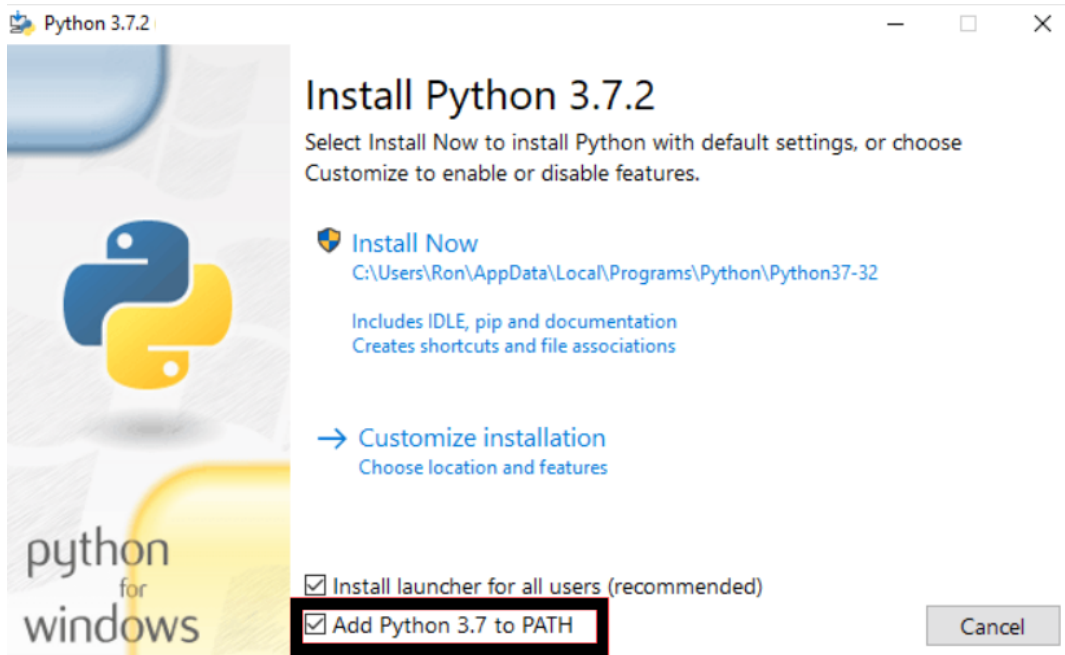
1- Installing Python 3.x.

Step 1: Please visit official Python website (<https://www.python.org>).

Step 2: Download and install the latest Python 3.x installer corresponding to your operating system. (64-bit version is recommended.)



Please note: for windows users, please check mark “Add Python 3.x to Path” on installation.



Step 3: check if Python is installed properly. Please open command prompt CMD (or terminal for MAC/Linux users) and enter the following command: **python --version**

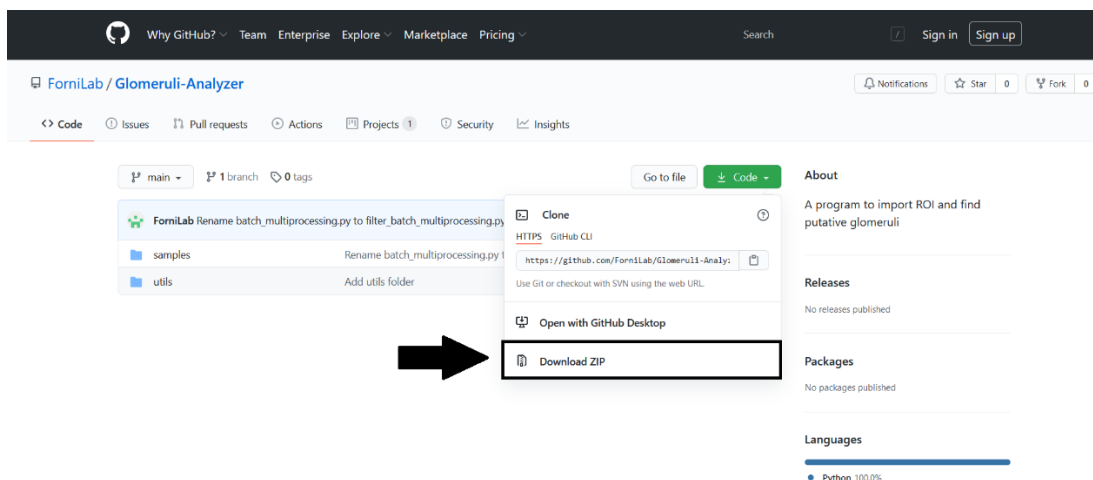
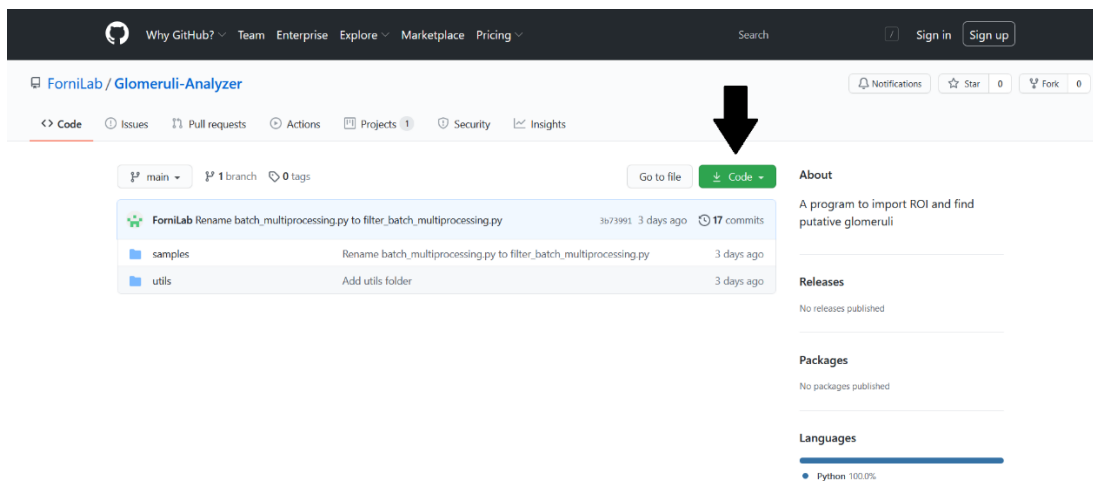
The result should be as follows: **Python 3.7.2**

Please note: if you got different version of Python on CMD/Terminal output, that means you have installed multiple Python engine and the default is the one that appeared in CMD/Terminal. It is strongly recommended to uninstall all existing Python engines and reinstall the desired one again.

2- Installing required Python packages.

All required packages to run the software properly are listed into a file called “requirements.txt”. This file should be available on repository.

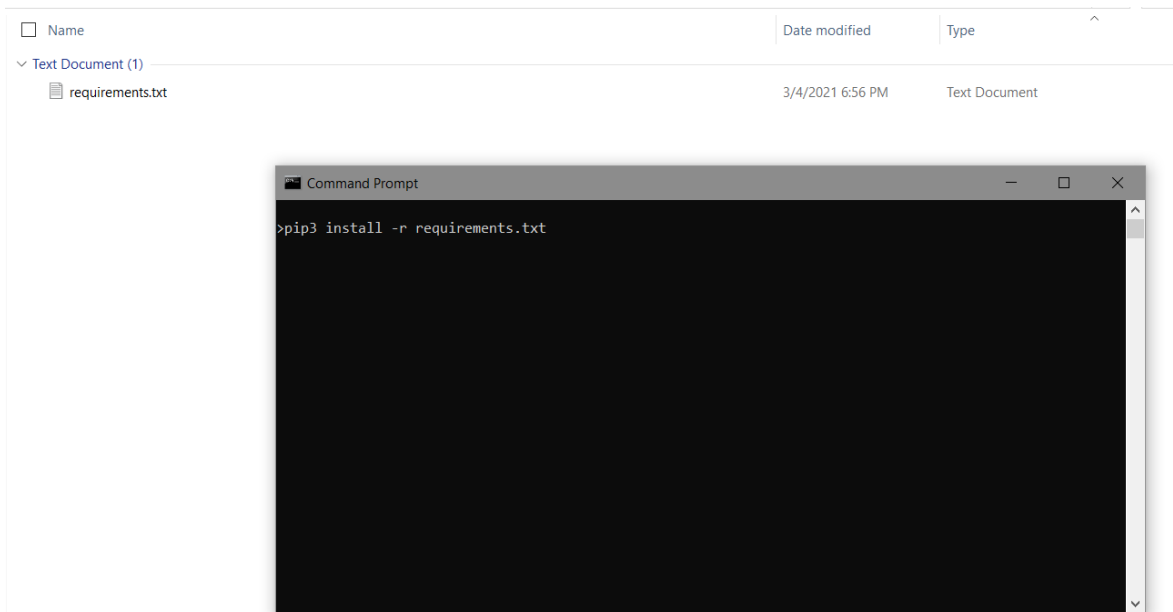
- a) In order to install packages, please visit “[Glomeruli-Analyzer](#)” project webpage download it as a zip file.



- b) Please extract downloaded zip file in a folder and follow navigate to this path to get “requirements.txt”:

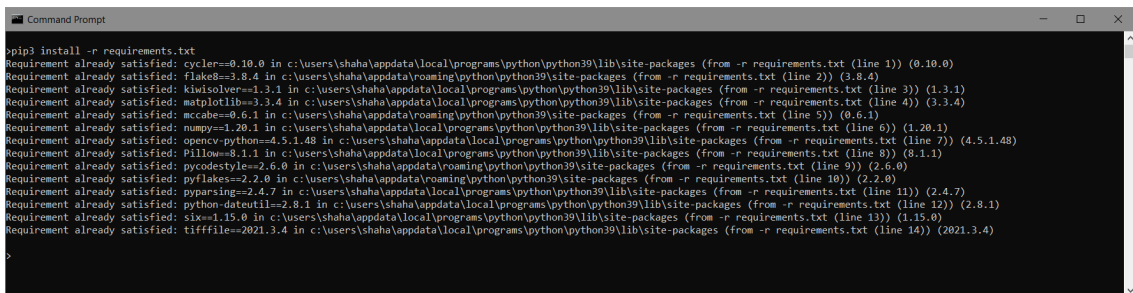
[Glomeruli-Analyzer](#) / [samples](#) / [filter_sample_code](#) /

Next, open CMD/Terminal in the corresponding folder that you have put the downloaded file and enter this code into CMD/Terminal: **pip3 install -r requirements.txt**



The required packages should start to install afterwards. Otherwise please check what packages are not able to be downloaded/installed and look for the error through python's forums to troubleshoot it.

If everything installed properly and you run the code again should receive the following message in CMD/Terminal: **Requirement already satisfied: ...**



How to run the program

There are various ways to run the program in different operating systems. In general, we divide it into two different methods:

1- Run by CMD/Terminal (multiprocessing)

Please download the whole folder as a zip file and extract it into a folder in your local machine.

Step1: put all images into “images” folder. (subfolders would be ignored)

Step2: change setting in “setting_batch.py” into your desired values.

Step3: run CMD/Terminal in the extracted folder and enter the following code:

Preferred for Windows/Linux users (multiprocessing)
python filter_batch_multiprocessing.py

Preferred for Windows/Linux/MAC users
python filter_batch_labview.py

Please wait until the process is completed.

Step4: check “output” folder for all analyzed images output. Each images analysis should be separated into folders.

Step5: there is an “output.txt” in “output” folder which include all parameters for all images. One may copy all data and paste it into an excel file.

2- Using compiled GUI or LabVIEW subVI.

In order to use compiled GUI please choose the corresponding file for your operating system and run the compiled standalone program.

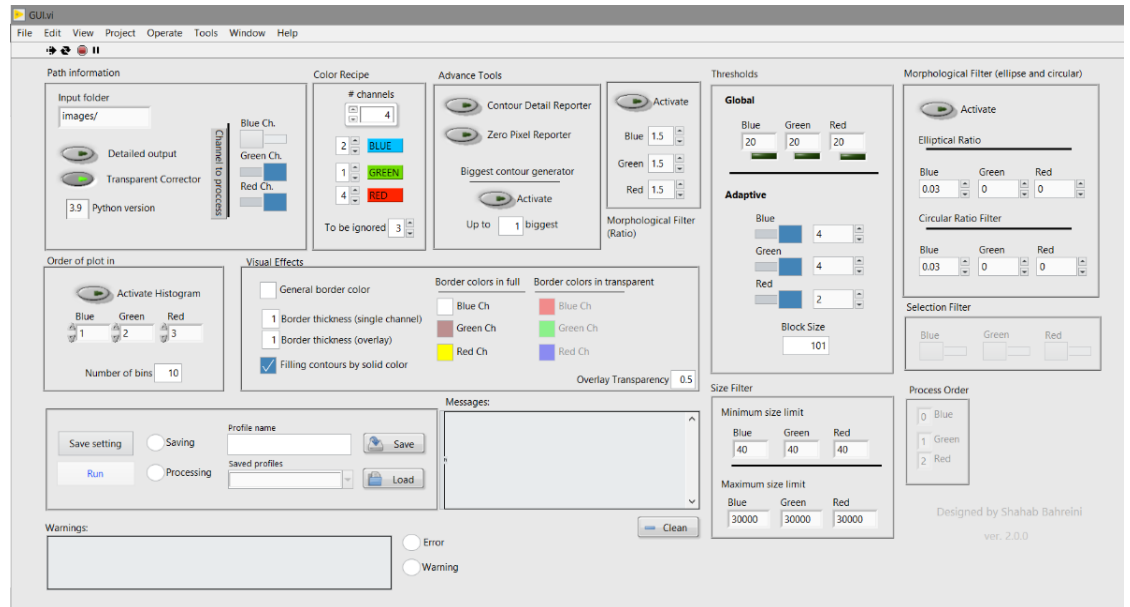
a) **(Recommended)** To use LabVIEW sub-VI:

- i. Please download, install, and activate LabVIEW community (above 2020) from its official website:

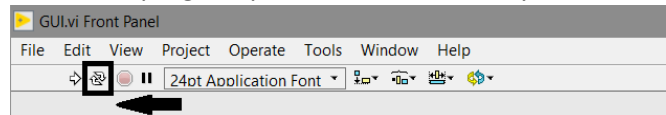
(<https://www.ni.com/en-us/support/downloads/software-products/download.labview.html>)

Please note: if you installed Python 3.x 64-bit you have to install LabVIEW 2020 64-bit, and vice versa for 32-bit version as well.

- ii. Open GUI.vi in LabVIEW and run it.



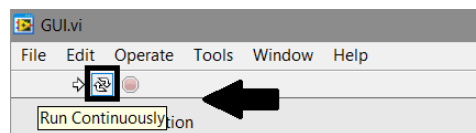
To run the program press “run continuously” on toolbar.



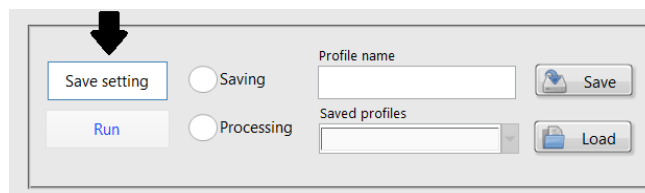
b) Run compiled version:

Windows OS: please run “Application.exe” and on the very left of the toolbar click “run continuously”.

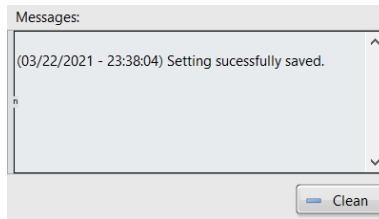
MAC OS: please run “Application-MAC” and on the very left of the toolbar click “run continuously”.



To process an RIO, you may choose your best setting by buttons and available controls and after all parameters are set, please click “Save setting”:

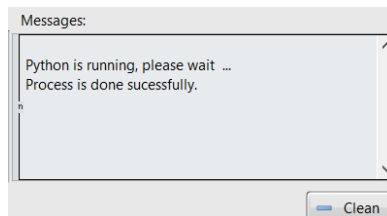


The following message should be appeared in message box. And in the program root folder “setting_batch.py” should be added/created at the same time.



Please note if “setting_batch.py” is an old one (check the modified time of the file), that means GUI did not run properly. Then please use another mentioned (1 or 2) method.

After saving setting, by clicking on “Run” button “outputs” folder should be updated with the last process images and the following message should be appeared there.



Please note if “outputs” folder did not update with new processed file, that means GUI did not run properly and it is highly recommended to run the program by Terminal/CMD. To do so, please open Terminal/CMD in the program root folder and run the following code:

python filter_batch_labview.py

In terminal you would see, total process time, number of files and some process information per each image as follows. Below is the result of processing red and green channel for “test.png” sample file which is available in folder “images”.

```
Command Prompt

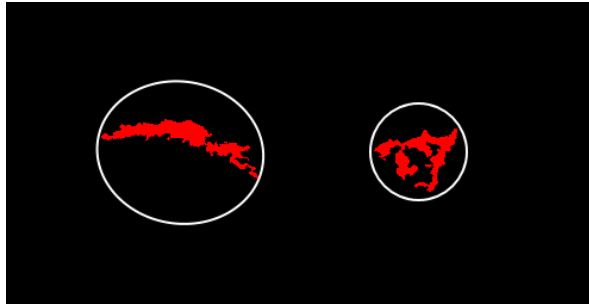
>python filter_batch_labview.py
Source is an PNG!
test.png
Image Information: -----
Image Dimension : (260, 601, 3)
Image Height : 260
Image Width : 601
Number of Channels : 3
-----
Max Pixel Value :255
Number of contours detected in G Channel = 671
@After 1st filter by size:48
Maximum size was:122806.5
Max Pixel Value :255
Number of contours detected in R Channel = 544
@After 1st filter by size:55
Maximum size was:126372.5
Successful!

1 files processed.

Duration: 0:00:00.187000
>
```

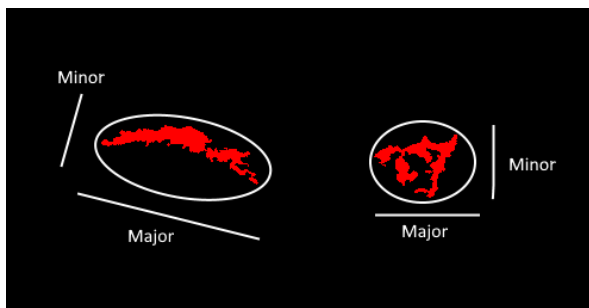

Filter tools

The goal of using morphological filters is to ignore axon like identified shapes or in other words set a limit of acceptance of elongated contours.



In circular filter, program encircles contours into a circle. By calculating the contour area and circle area the following ratio is determined: $Size/\pi r^2$

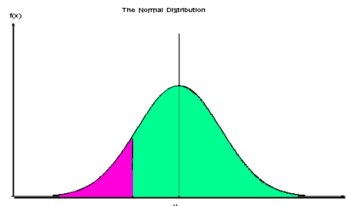
This ratio indicates how thin and elongated is the contour. So, for axon like contours this ratio is low. Circular filter tries to put a limit of acceptance for this ratio to ignore axon like contours.



Ellipse filter inherited the same method of circular filter; however, it encircles contours with an ellipse. The $minor\ axes/major\ axes$ is calculated to check how elongated is the contour. Smaller ratio means more elongated shape.

In both cases acceptable range of circular/ellipse ratio is defined as below:

Acceptable range: $[threshold, 1]$



Ratio filter also is another tool to help removing axon like shapes with a different method. In this method program computes the following ratio of each contour: $perimeter/area$

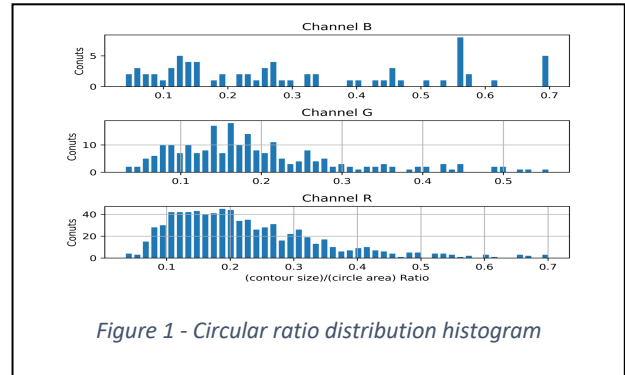
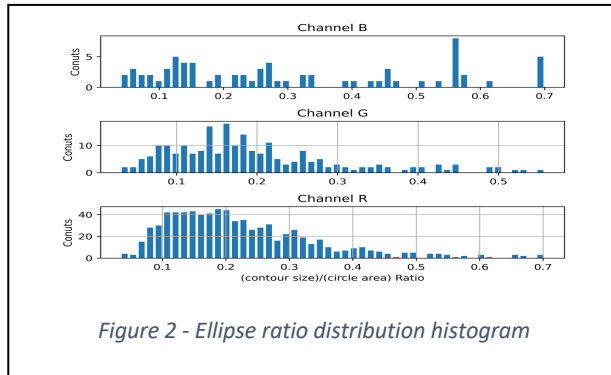
The larger ratio indicates the more snakier shapes (or contours with more thin wire like branches). To set a limit of acceptance, by setting a positive number (λ), program ignores all contours that has a ratio below the following range:

$$acceptable\ ratio < (maximum\ ratio - \lambda\sigma)$$

Here σ indicates standard deviation of ratio distribution.

Histogram

In order to determine threshold ratio in morphological filters, histogram of ratio distribution helps. This feature allow program to generate histogram of ratio distribution of circular and ellipse filters. Below is a sample of generated histogram for RGB channels (see figure 1,2).



Size limit filter

The goal of this tool is to eliminate miniscule identified contours. Figure 3 is a sample of processed raw ROI image which contains a large number of tiny patterns.

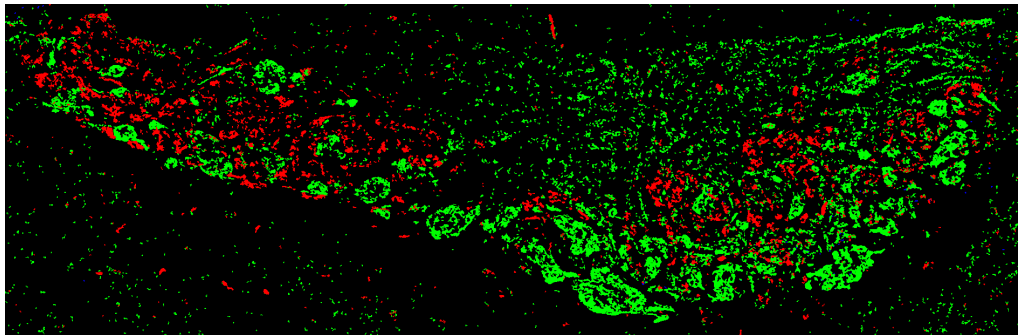


Figure 3 - sample of output for a raw ROI. Too many tiny patterns are identified in image.

However, set a size limit threshold helps to purify program output (see Figure 4).

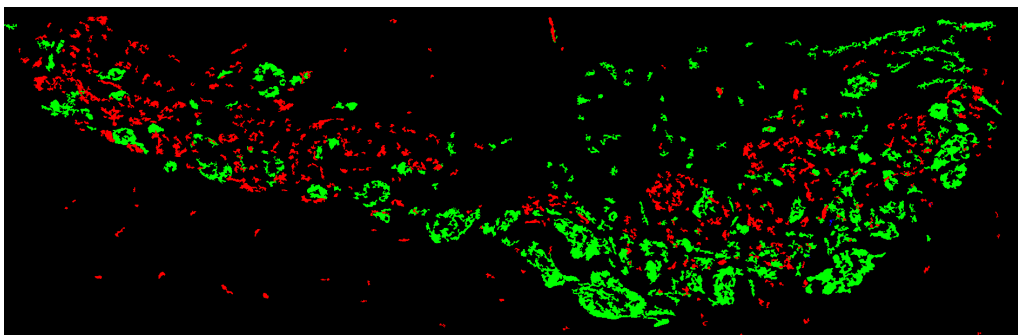


Figure 4 - Set size limit 10, 40 and 10 for RGB channels respectively helps to eliminate small redundant patterns.

Technical Setting Description

Path Information

Name	Description	Value	Available in GUI (graphical interface)	Available in file (manual edit)
Input folder	Folder name which contains ROIs. (no path is acceptable, just the name + "/")	Name + "/"	✓	✓
Detailed output	Enable/Disable contour details per images. This option allows program to generate full detailed result per ROI.	True/False	✓	✓
Transparent corrector	Removes any available solid background from ROIs.	True/False	✓	✓
Channel process	Determines which channel should be processed.	True/False	✓	✓

Advanced Tools

Name	Description	Value	Available in GUI (graphical interface)	Available in file (manual edit)
Contour detail reporter	<u>For development purpose</u> – generates all points in contours in OpenCV Raw format.	True/False	✓	✓
Zero-pixel reporter	<u>For development purpose</u> – reports how many pixels with absolute value 0 are exist in ROIs.	True/False	✓	✓
Biggest contour generator	Generate the most biggest identified contours in the separated image output.	True/False	✓	✓
Number of most biggest contours	Determines the number of biggest contours that should be included in output.	Integer > 0	✓	✓

Python version	Installed and default python version. Just first two values are acceptable. For example, 3.7 or 3.9 not 3.7.2 or 3.9.1	3.x	✓	✗
----------------	--	-----	---	---

Process order	Blue Ch.	<u>For development purpose</u> – determines which channel processed first and so on.	0 ,1 ,2	✓	✓
	Green Ch.				
	Red Ch.				

Thresholds

Name		Description	Value	Available in GUI (graphical interface)	Available in file (manual edit)
Global	Blue	Activates when adaptive threshold is disabled. A small LED indicator is below each channel box which turns on when global threshold activates for each channel.	Integer > 0	✓	✓
	Green				
	Red				
Adaptive	Blue	Enable/Disable adaptive threshold per channel.	Integer > 0	✓	✓
	Green				
	Red				
Block size		Determines adaptive threshold block size	Odd Integer > 8	✓	✓

Morphological Filter (ellipse and circular)

Name		Description	Value	Available in GUI (graphical interface)	Available in file (manual edit)
Activate		Enable/Disable both ellipse and circular filters	True/False	✓	✓
Elliptical Ratio Filter	Blue	This filter tries to encircle each contour by an ellipse. It determines the ratio of (minor axis)/(major axis). Next, filters out all those contours which has smaller value than computed ratio.	> 0	✓	✓
	Green				
	Red				
Circular Ratio Filter	Blue	This filter tries to encircle each contour by a circle. For circular filter it computes ratio of (contour area)/(circle area). Next, filters out all those contours which has smaller value than computed ratio.	> 0	✓	✓
	Green				
	Red				

Morphological Filter (Ratio)

Name		Description	Value	Available in GUI (graphical interface)	Available in file (manual edit)
Activate		Enable/Disable both ellipse and circular filters	True/False	✓	✓
Ratio Filter	Blue	This filter find the ration of (contour perimeter)/(contour size) and ignores those who have ratio above (maximum ratio – [entered value]*std)	> 0	✓	✓
	Green				
	Red				

Selection Filter

Name		Description	Value	Available in GUI (graphical interface)	Available in file (manual edit)
Selection Filter	Blue	If activated labels contours and asks for contour number to remove. The contour numbers are supposed to be in a text file with the name of "filter-number-(R/G/B).txt" for each channel per image separately. It is not designed for batch process. (Disabled in current version)	True/False	✓	✓
	Green				
	Red				

Size Filter

Name		Description	Value	Available in GUI (graphical interface)	Available in file (manual edit)
Minimum limit size	Blue	Set the minimum acceptable contour size per channels.	Integer > 0	✓	✓
	Green				
	Red				
Maximum limit size	Blue	Set the maximum acceptable contour size per channels.	Integer > 0	✓	✓
	Green				
	Red				

Visual Effects

Name		Description	Value	Available in GUI (graphical interface)	Available in file (manual edit)
General border color		Contour color border for isolated channel contour maps	RGB Color Code	✓	✓
Border thickness (single channel)		Contour border thickness in isolated channel contour maps	Integer > 0	✓	✓
Border thickness (overlay)		Contour border thickness in transparent overlay contour maps	Integer > 0	✓	✓
Filling contours by solid color		Enable/Disable filling contours with solid color (the same color as border).	True/False	✓	✓
Overlay Transparency		Sets transparency factor. 1 means totally transparent, and 0 means opaque.	0.0 – 1.0	✓	✓
Border colors in full map	Blue Ch.	Determines the contour border color per channel in full contour map	RGB Color Code	✓	✓
	Green Ch.				
	Red Ch.				
	Ch.				

Border colors in transparent map	Blue Ch.	Determines the contour border color per channel in transparent contour map	RGB Color Code	✓	✓
	Green Ch.				
	Red Ch.				

Order of plot in histogram

Name		Description	Value	Available in GUI (graphical interface)	Available in file (manual edit)
Activate Histogram		Enable/Disable generating histogram for elliptical and circular filter ratios.	True/False	✓	✓
Channel Order	Blue	Order channels for generated histogram. 1 has the most priority.	1, 2, 3	✓	✓
	Green				
	Red				
Number of bins		Determines the number of bins for generating histogram.	Integer > 0	✓	✓

Color Recipe

Name	Description	Value	Available in GUI (graphical interface)	Available in file (manual edit)
# channels	Determines how many channels are integrated into ROI. (In this version, program just accept 3 or 4)	3, 4	✓	✓
BLUE	The channel number to set as blue channel. Maximum possible value is the ROI channel number.	Integers < # channels	✓	✓
GREEN	The channel number to set as green channel. Maximum possible value is the ROI channel number.	Integers < # channels	✓	✓
RED	The channel number to set as red channel. Maximum possible value is the ROI channel number.	Integers < # channels	✓	✓
To be ignored	Channel number which is supposed to be ignored in analysis.	Integers < # channels	✓	✓

Save & Run

Name	Description	Value	Available in GUI <small>(graphical interface)</small>	Available in file <small>(manual edit)</small>
Save setting	Saves the current setting as the one which program use it.	-	✓	✓
Run	Run the process. Please wait until processing LED turns off. Program runs with the last saved setting.	-	✓	✓
Profile name	The name of current setting that wants to be saved and be loaded in future.	Integers < # channels	✓	✓
Save	Saves the current values into a file which can be loaded in future. All profiles will save into “profiles” folder with “.ini” file extension.	Integers < # channels	✓	✓
Saved profiles	This drop-down list shows all available profile to choose and load.	Integers < # channels	✓	✓
Load	Loads the selected profile from drop-down list.			