



UNIVERSITY OF TRENTO
DEPARTMENT OF PHYSICS

Laboratory Report: Segmentation of Neuron Images

Leonardo Cattarin

Laboratory of Advanced Photonics
Paolo Bettotti, Davide Bazzanella

1 Introduction

1.1 Optical Microscopy

Optical Microscopy is one of the most established microscopy techniques, used in a great variety of fields and on different subjects. The imaging and analysis of biological tissue is one of the most important of such applications. Confocal optical microscopy is one of the most used methods, but is unfortunately limited by the wave nature of light itself. The image formation process requires diffraction to happen, but due to the practically finite aperture of the lenses used, part of the light emitted from the sources of interest does not reach the lenses used and does not contribute to the image reconstruction. Therefore, confocal microscopy is limited by the so-called Rayleigh limit (Equation 1.1), which in point-like objects usually appears as the "Airy pattern".

$$d_R = \frac{0.61\lambda}{n \sin \alpha} \quad (1.1)$$

where λ is the light wavelength, which cannot be decreased at will, n is the refractive index of the medium where the target is located, and α is half the angle of aperture of the objective lens used in the configuration. Equivalently, lenses behave as low-pass filters for the spatial frequency content of images. Effectively, this limit make objects under this length undistinguishable.

A possibility to avoid such problem is the super-resolution technique called Structured Illumination Microscopy (SIM) [1] based on the idea of illuminating the sample with non-uniform light patterns. Non-uniform illumination patterns allow to explore broader sections of the spatial frequencies space, and doing so repeatedly allows to overcome the Rayleigh limit. A common light structuring configuration is based on the usage of a "Nipkov" disk, a disk with small pinholes which is made spinning. Multiple images are taken at appropriately synchronized times, and they are used to reconstruct the final image.

δx	δz	bit depth	Resolution ratio	Magnification
0.065 μm	0.25 μm	16	1400 \times 1400	100x

Table 1: Microscopy parameters. From left to right: pixel size, z-stack step, single channel bit depth, image resolution, Magnification

The images used here come from two different Neuron samples, and are originally acquired both in Confocal and SIM configurations. The images are further obtained for two different kind of fluorophores, RFP and CY5, and at various focal lengths. The image parameters are presented in Table 1. They are also acquired also at different focal lengths, or "slices". This produces images where different parts of the sample are more in-focus, leading to images with different intensities spatial distributions .

Some images are presented in Figure 2, Figure 1 and Figure 3.

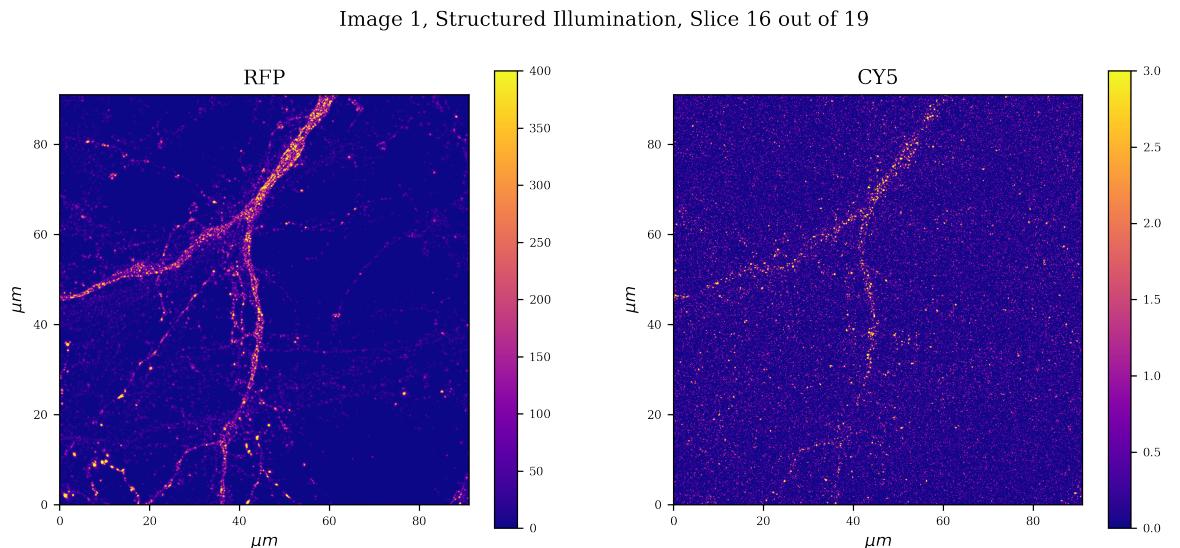


Figure 1: Image 1, Structured Illumination taken with different fluorophores

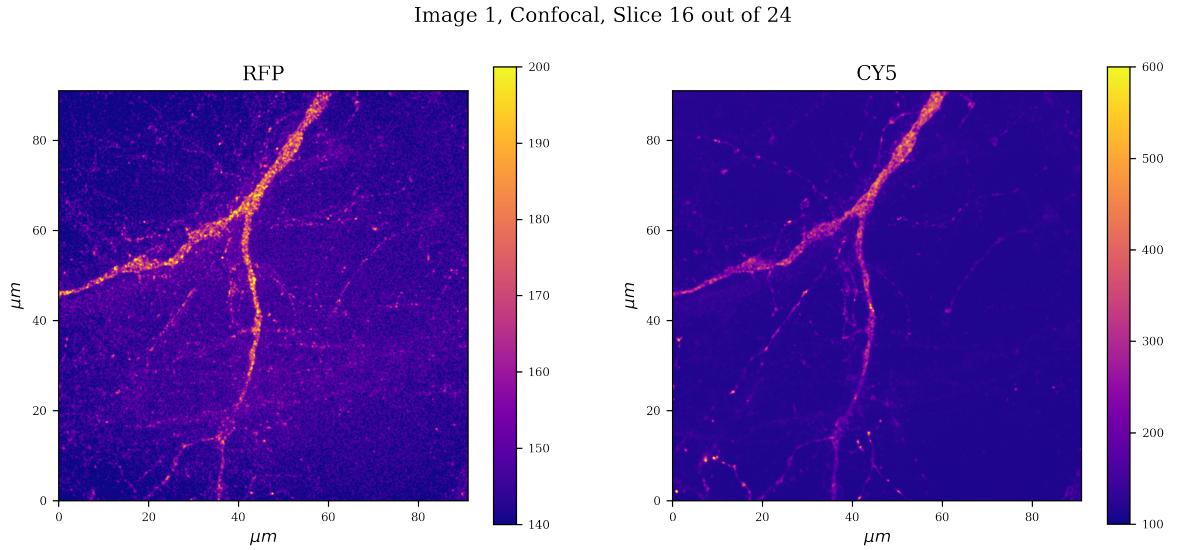


Figure 2: Image 1, Confoca, taken with different fluorphores

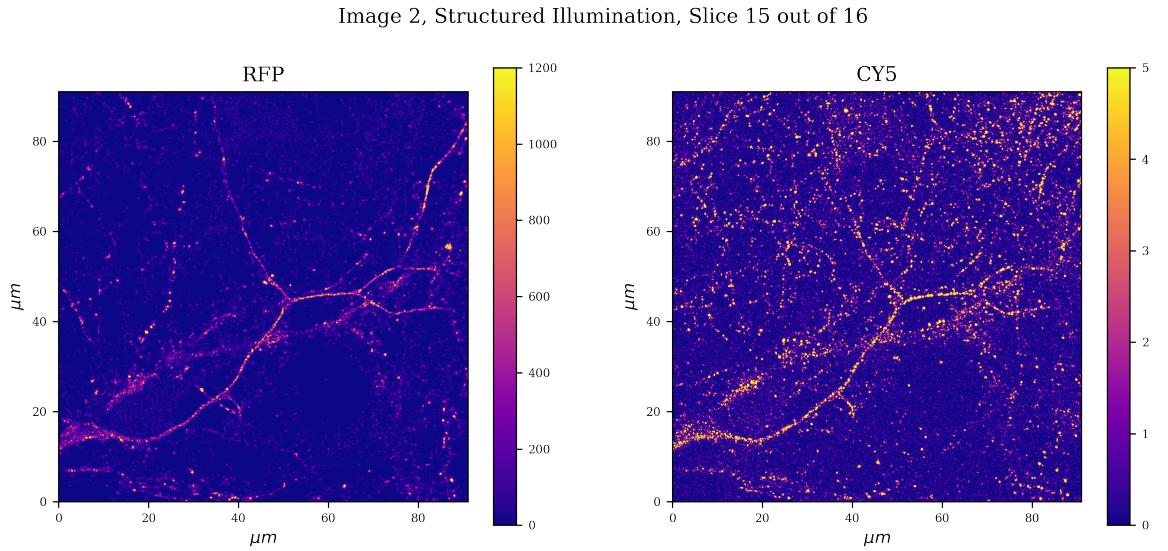


Figure 3: Image 2, Structured illumination taken with different fluorphores

As a first note, in structured illumination images CY5 usually offer worst signal to noise ratios, while images with RFP phluorophores offer stronger signal intensities.

In the end, the images considered best to perform analysis were the structured illumination RFP ones, and the slices number used were 12-18 for Image 1 and 13-15 for Image 2.

2 Image Analysis

Monochromatic (single-channel) digital images are here represented as matrices $\{I_{ij}\}$ with $i = 1, \dots, N$, $j = 1, \dots, M$, $(N \times M)$ being the image size in number of pixels.

Once the images are acquired and selected, a possible task is to segment (or partition) the image in multiple domains to isolate the main connected components of the neuron (the axon) from the background noise and the secondary features (such as dendrites).

This operation is here performed by using some filtering techniques described below, such as threshold filtering, morphological filtering and connected components labeling. First, these "building block" methods are described one by one, and then the complete filtering procedures are illustrated, usually composed by multiple iterations of the basic filters described above. The methods described are implemented or used in Python, using the libraries OpenCV, Numpy and Scipy [2]. The code can be found on GitHub [3].

2.1 Basic Algorithms and Filters used

2.1.1 Thresholding

Thresholding methods aim at filtering image features based on their intensity levels. Various kind of thresholding are possible, but here only binary thresholding is used. Given an initial single-channel image, each pixel I_{ij} is kept non-zero or set to a selected value only if its intensity is above a value ξ_{th} . The obtained object is either a binary mask usable for further processing (where the pixels can only assume 0 or 1) or a new image.

2.1.2 Erosion

The erosion filter works by convolving a Kernel Matrix over the image, and by substituting to the "anchor point" the local minima found in the kernel-covered region. Given the kernel matrix K_e of chosen dimensions $(n \times m)$, for each pixel (anchor point) I_{ij} of position (i, j) , the pixels at $(i + k, i + l)$ are considered, with $-n/2 < k < n/2$, $-m/2 < l < m/2$. The original pixel is then substituted with the local minima found for (k, l) values for which $(K_e)_{k,l} \neq 0$

For binary mask images, this is equivalent to say that a pixel is set to 0 if at least another one is zero within the kernel size. Practically, it sets to zero or "erodes" parts of the image which are more marginal respect to relatively big bodies. Objects roughly smaller than the kernel size are expected to be erased.

2.1.3 Dilation

Similar to erosion, substitutes pixel values with the local maxima found within kernel nonzero regions. For binary masks, sets a pixel to 1 if at least another one is 1 within the kernel K_d size. It "dilates" objects or it helps connect previously disconnected ones. Holes roughly smaller than the kernel size in nonzero regions ("objects") are expected to be closed.

2.1.4 Opening

An erosion followed by a dilation. Useful to remove small outlying objects, such as noise or small unwanted features. The subsequent dilation is used to restore the boundaries of relatively big objects, which are altered by erosion. Small deleted features do not reappear.

2.1.5 Closing

Consecutive action of a dilation and an erosion. Used to close small holes while not altering macroscopic boundaries and shape of objects.

2.1.6 Connected components labeling

An algorithm based on "Spaghetti Labeling" [4] used to identify image components which are connected. The algorithm reformulates the image as a graph and identifies his connected subcomponents. The algorithm used assumes 8-way connectivity: that is, each pixel is a node connected to 8 other ones, which are the nearest neighbours pixels both horizontally, vertically and diagonally.

The final object returned by `connectedComponentsWithStats` is an image/matrix where each pixel is labeled by an integer according to the identified component it belongs. Along with this other informations are returned in a tuple, such as the component sizes, centroids and label numbers.

This algorithm is used here by considering only the biggest obtained connected component (a wrapping function `get_biggest_object` does this).

2.2 Image Processing

A first method is here presented, which is able to isolate a smaller component identified as the axon, but loses a lot of smaller features which could be in principle of interest. To try to recover more informations, first the process is reiterated for multiple image slices (corresponding to different focal planes) and the obtained results are all projected together. Since this method is not able to solve the issue (some features are still left out), a new second method is developed with the focus of keeping more details from the original image.

2.2.1 Method 1

The first method consists in 4 steps:

- Simple Thresholding
- Closing
- Opening
- Biggest connected component retrieval

The initial thresholding is used to filter out trivial background noise at intensities low enough such that it can be distinguished from real features. The closing process is used to first close holes in the axon and with some dendrites. It must be noted that the kernel size must be big enough to connect relevant components but small enough to not connect with noise and spurious emission centers. The opening process is then used to clean from noise and unwanted features. Again, also in this case the kernel choice is a trade-off between risking to remove relevant features which are too small and not being able to clean up noise components or unwanted features due to a kernel size too small. The connected component analysis is then used to retain only the biggest segment of the image, from which the secondary features were removed by previous steps. This final object can be roughly identified as the Neuron axon.

Figure 4 and Figure 5 show the result of Method 1 over a single slice. Figure 6 Shows the binary masks corresponding to the various steps.

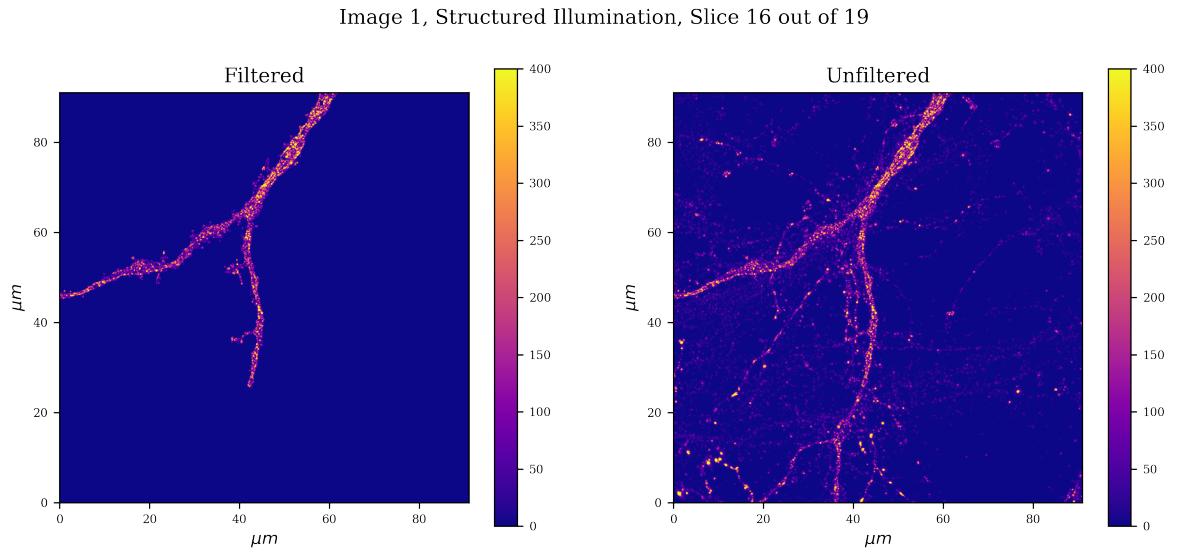


Figure 4: Image 1, Filtered using Method 1

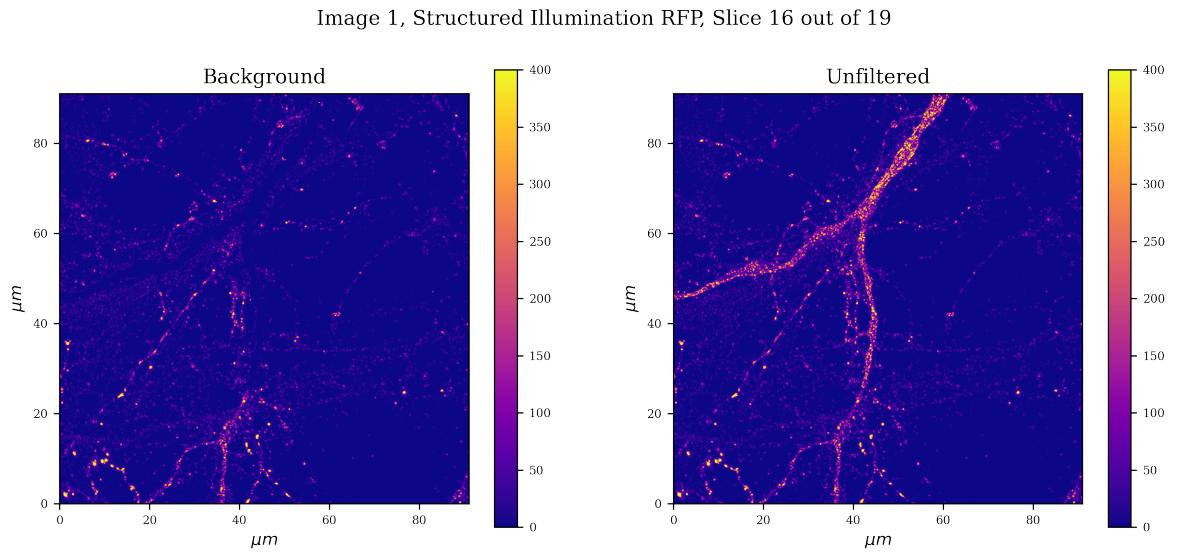


Figure 5: Image 1, Filtered using Method 1

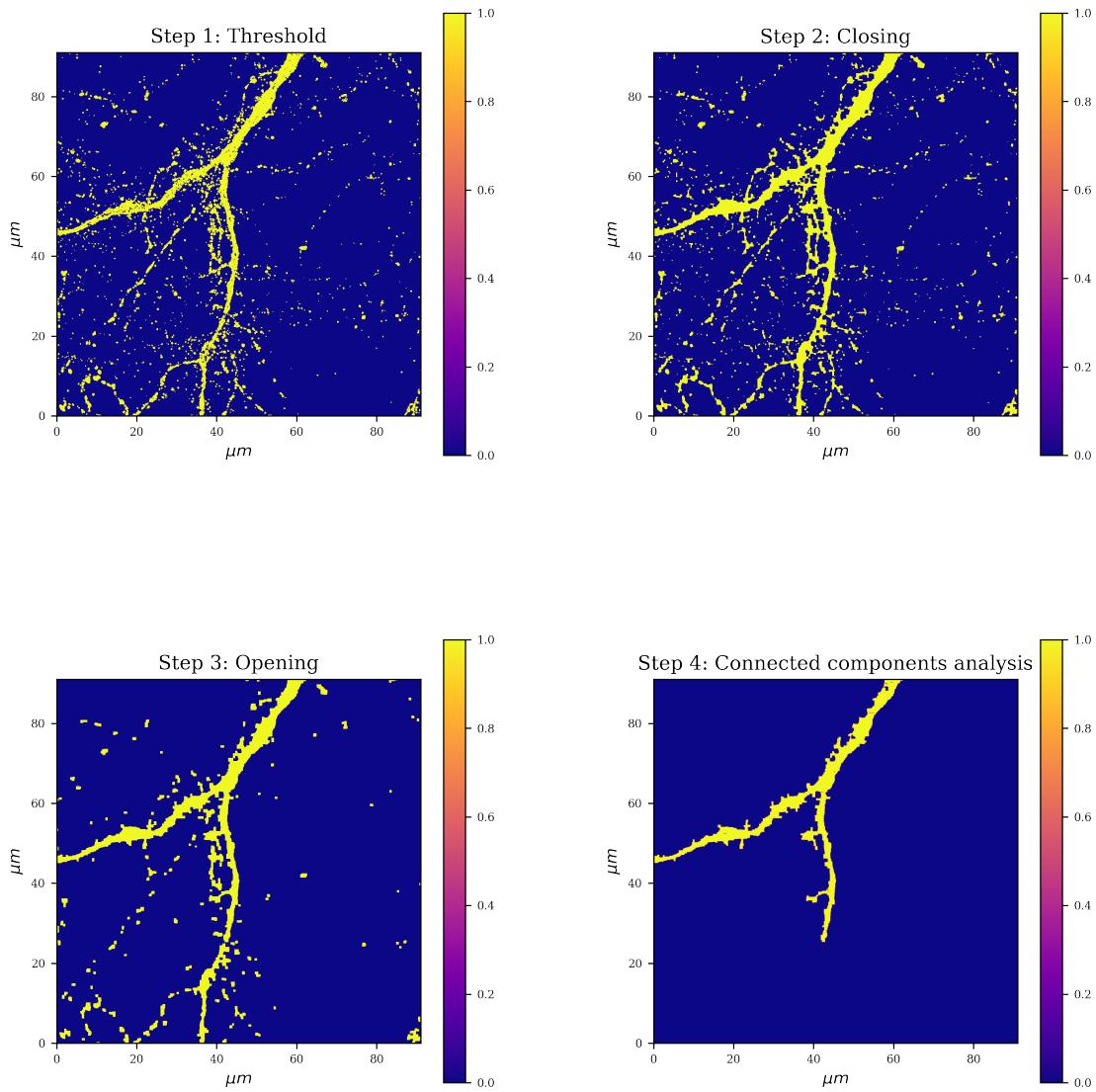
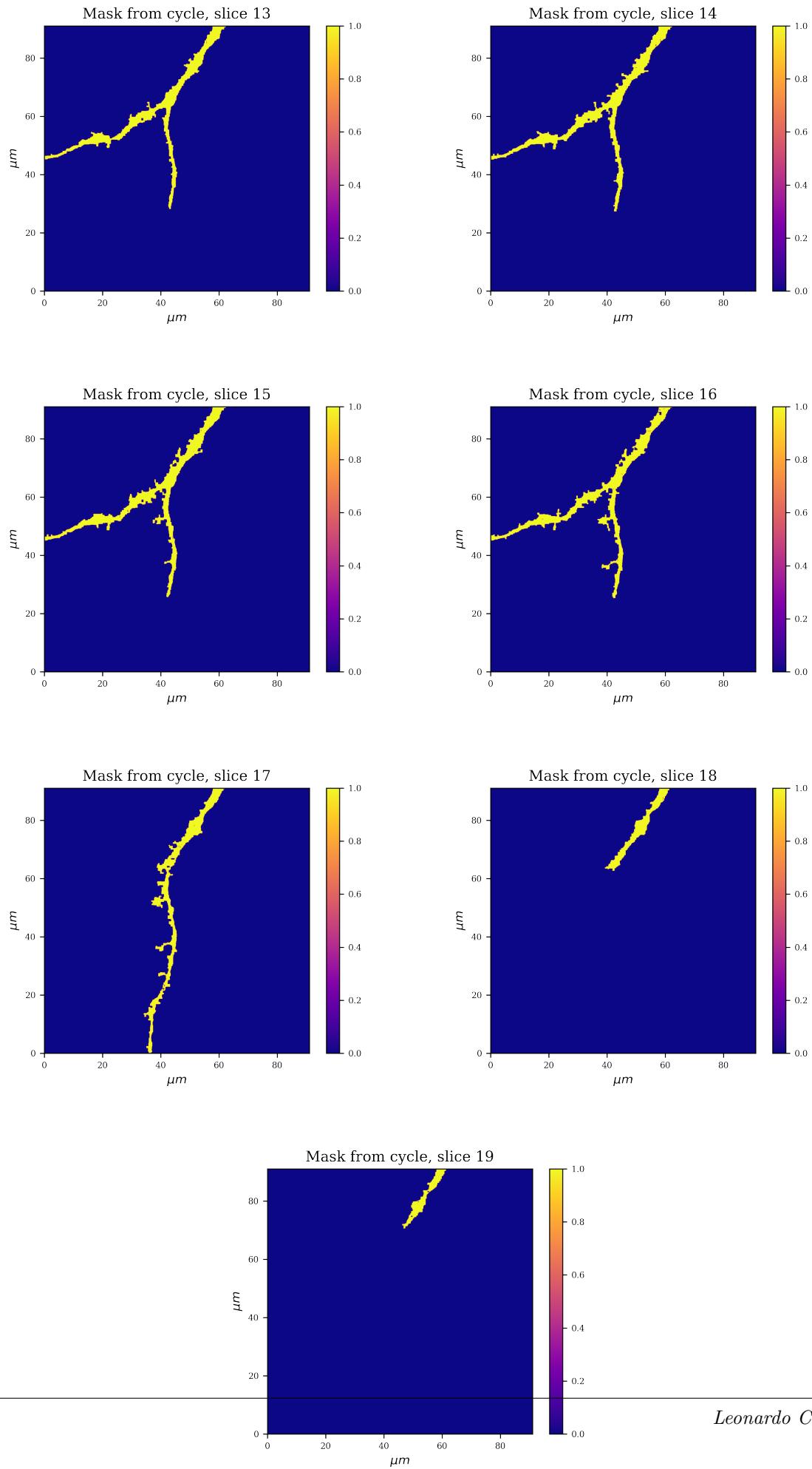


Figure 6: Filtering steps for Method 1.

The whole process is also iterated on multiple slices (focal planes), as shown in Figure 7. The final results are shown in Figure 8 and Figure 9.



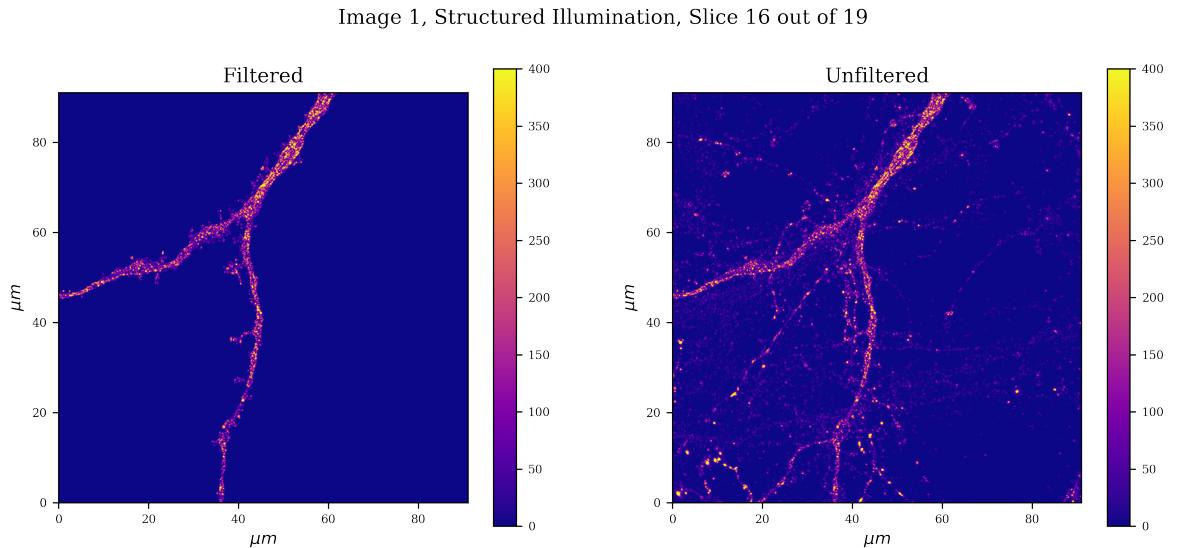


Figure 8: Image 1, Filtered using Method 1 over multiple slices

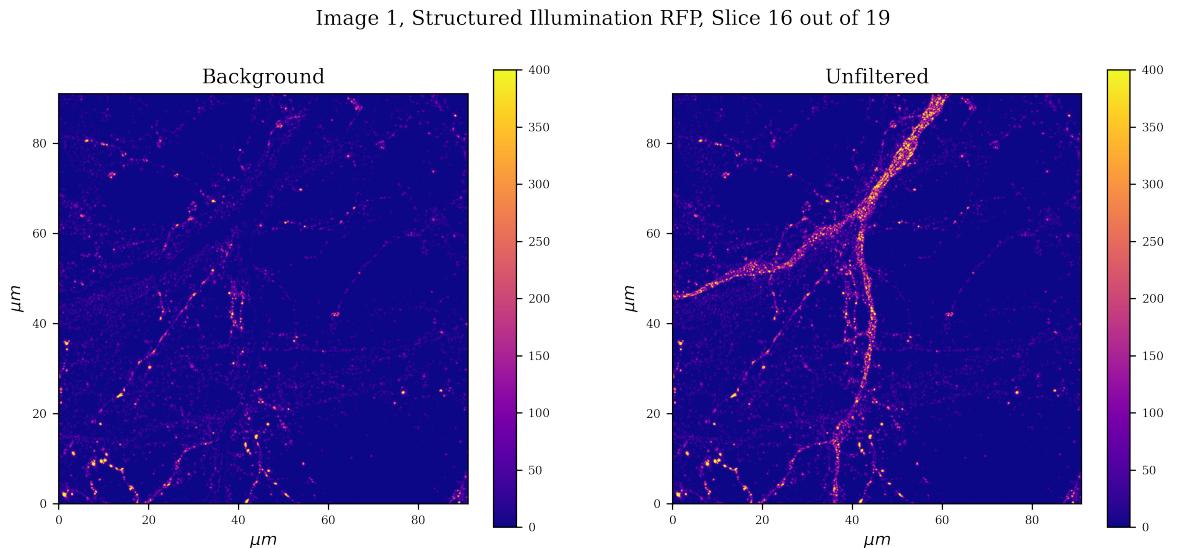


Figure 9: Image 1, Filtered using Method 1 over multiple slices

The effect of a modified version of method 1 is also shown for Image 2 in Figure 10 and Figure 11, and the process steps in Figure 12

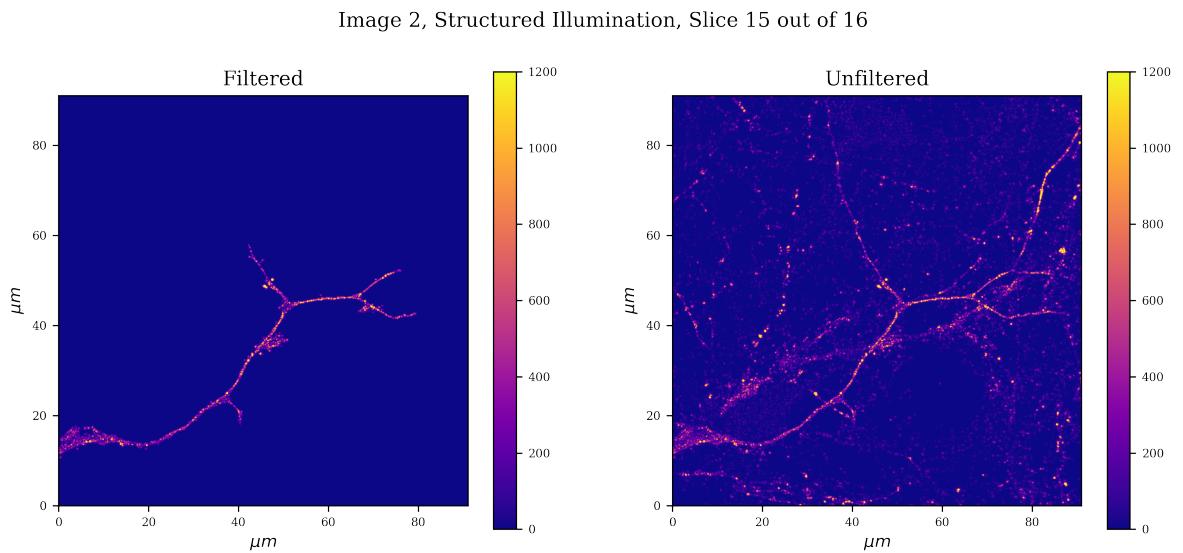


Figure 10: Image 2, Filtered using Method 1

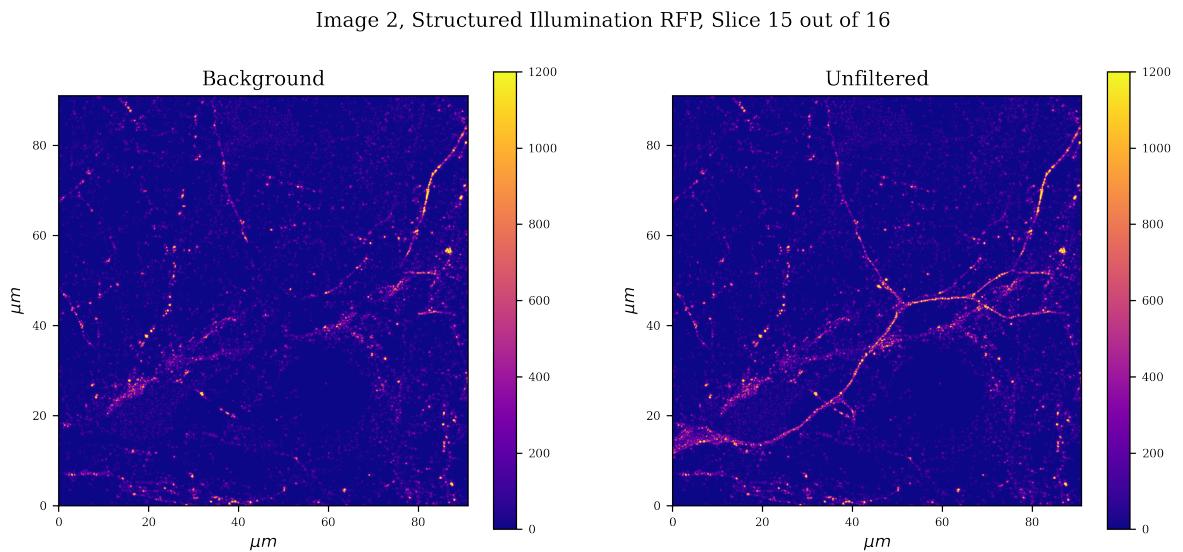


Figure 11: Image 2 Background, Filtered using Method 1

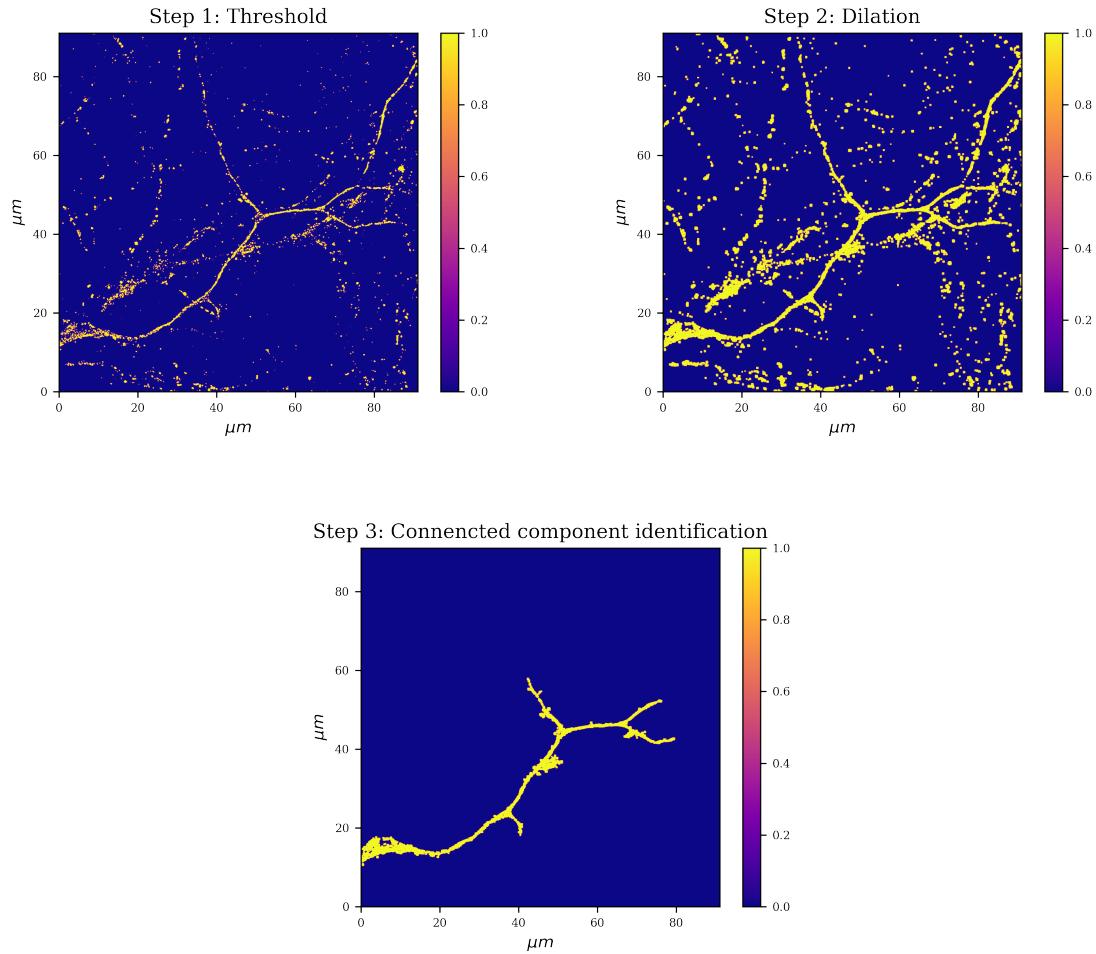


Figure 12: Filtering steps for Method 1, Image 2.

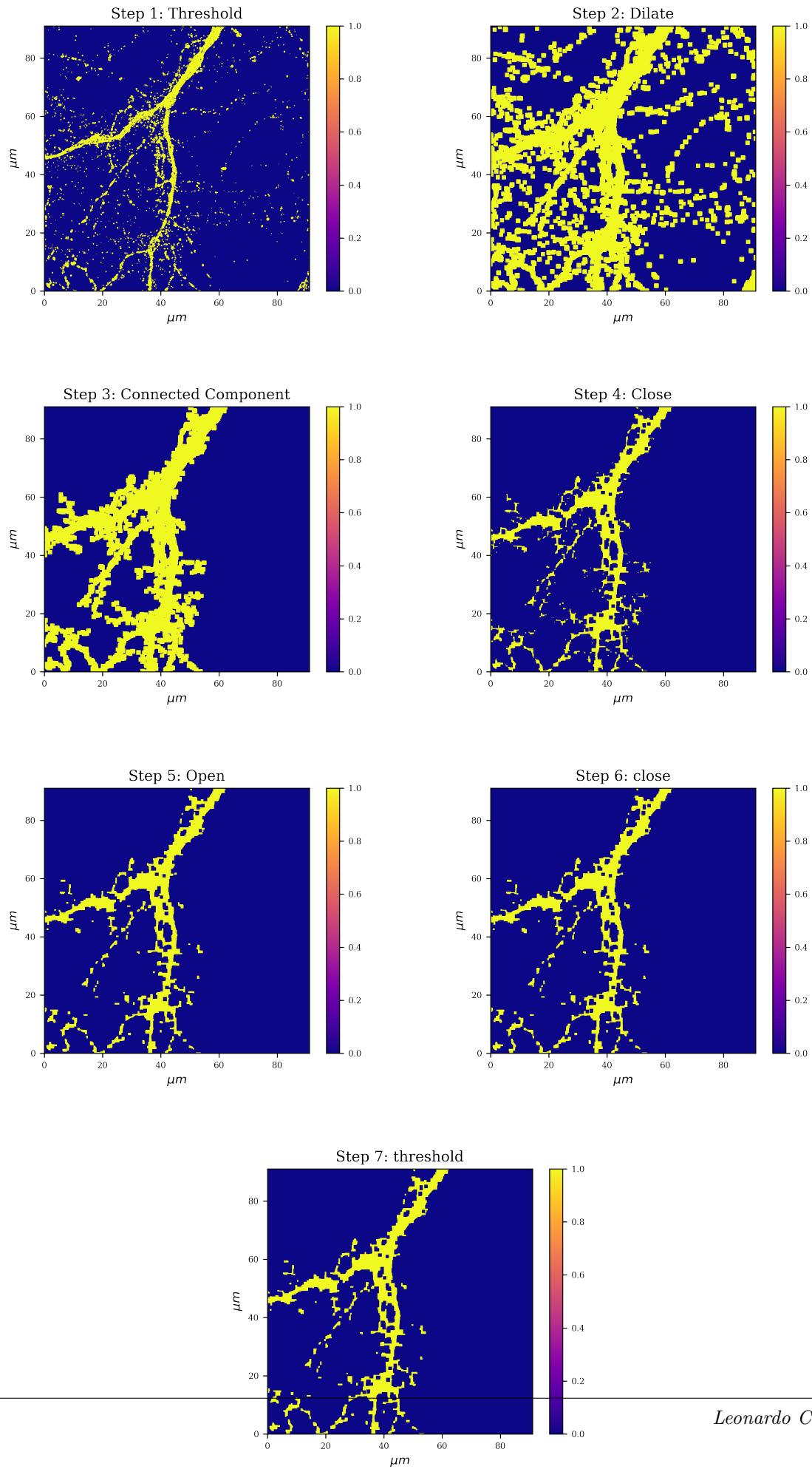
2.2.2 Method 2

The Second method consists in 7 steps:

- Simple Thresholding
- Dilate
- Connected component isolation
- Erode
- Open
- Close
- Threshold

The basic idea is, after the basic threshold filtering, to perform an initial dilation used to "expand" the most important components. This in turn helps to connect elements that previously risked to remain isolated. The biggest connected component is then isolated and the obtained mask is then eroded back as in a closure. The image is then refined by opening and closing with small kernel sizes, since sizes too big risk to lose the features acquired.

Figure 13 shows the masks corresponding to the various steps.



The final result is presented in Figure 14, Figure 15

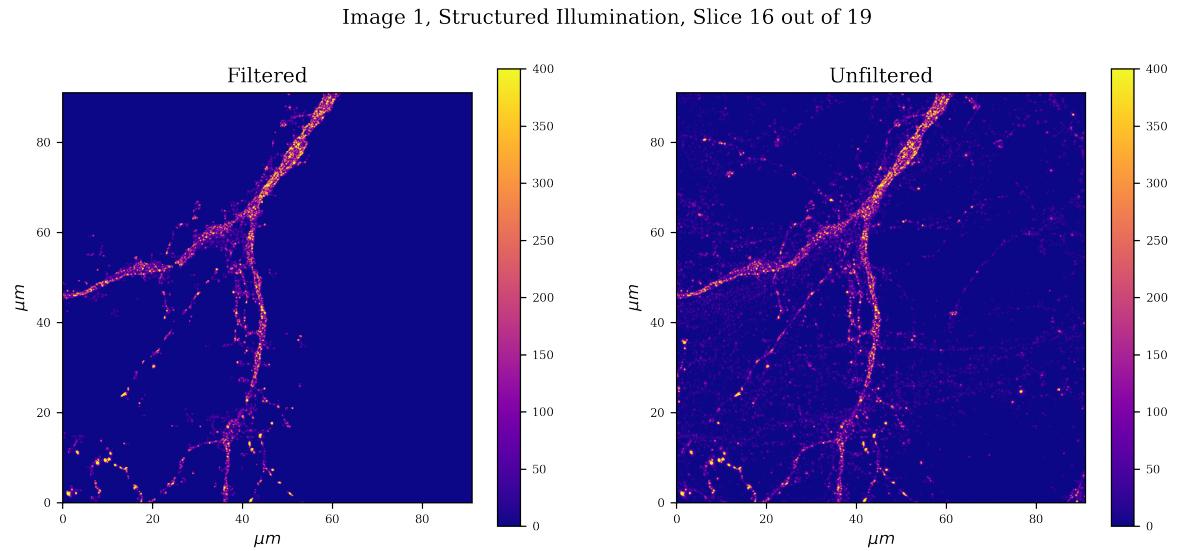


Figure 14: Image 1, Filtered using Method 2

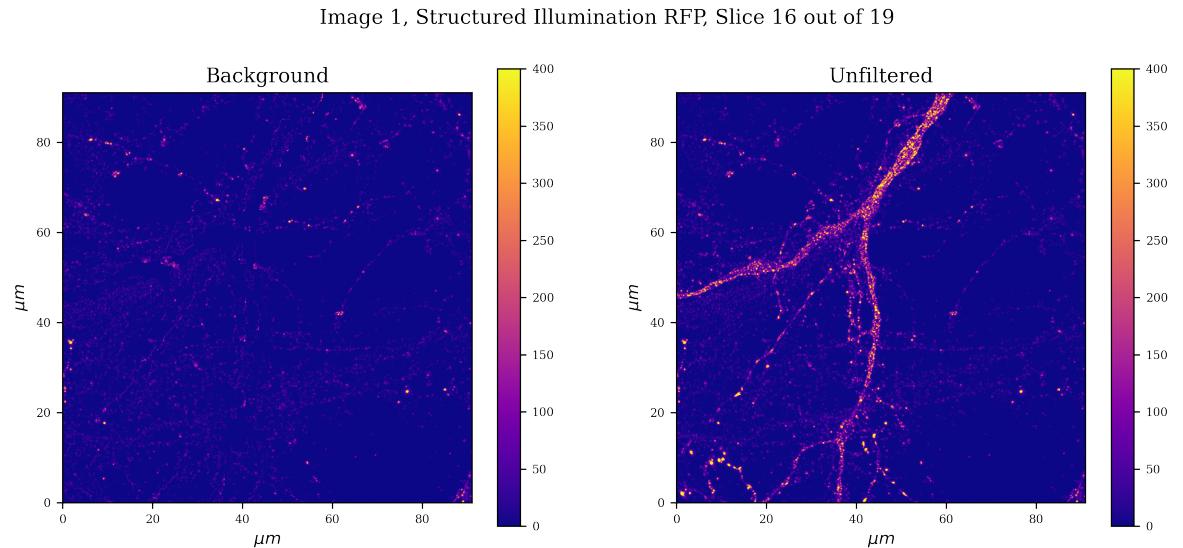


Figure 15: Image 1, Filtered using Method 2

References

- ¹K. Ulrich, *Fluorescence Microscopy From Principles to Biological Applications*.
- ²*OpenCV python Documentation*, https://docs.opencv.org/4.x/d2/d96/tutorial_py_table_of_contents_imgproc.html.
- ³L. Cattarin, *photronics final report*, <https://github.com/leonardocattarin/NeuronConnectivity>, 2022.

⁴B. Federico, A. Stefano, B. Lorenzo and G. Costantino, *Spaghetti Labeling: Directed Acyclic Graphs for Block-Based Connected Components Labeling*, 2019.