**Team-12**

**Neuron Classification in ImageJ**

Team Members:

Vijaya Lakshmi Boyina

Deepa Challa

Bhavani Duggineni

**Abstract**

Neuron which are captured in multiple frames needs to identified. A person cannot find Neuron characteristics on Simply looking those frames as these frames have different information store in different frames. imageJ tool provides different tools to process these types of images. Different Processes involved are 3D Filtering, Converting Image to Binary,Watershed algorithms, Particle Analyzer ,Level sets Segmentation. Segmentations identifies the neurons based on the seed point. Centroid is used as seed point . In Seed Growing Mechanism the neuron are thresholded based on a particular threshold range. 3D Object Counter finds the Surface area of neurons and Volume of Neurons.

**Process**

The approach we have applied while developing this plugin for ImageJ tool in java are as follows:

1. Storing all the images in a stack
2. As images may be noisy , so a Filter is applied.
3. After that the Image is converted to 8 bit binary image.
4. Then we set a Proper Threshold and apply watershed segmentation to identify cell location
5. Then Image is analysed using Particle analyzer and a table with cell locations is produced.
6. After Converting Image to Binary Then we apply level Set Algorithm which based on Seed Growing.
7. A Projector is applied to combine all images to produce a 3D image.
8. Then Different processes are used to find Surface area and Volume.

**Algorithm Design**

While Developing this Plugin for Neuron Segmentation Mainly Two algorithms comes in use:

1. Watershed Algorithm
2. Level Set

The other algorithms used are :

1. For filtering the image 3D Filter Algorithm is Applied

All the above Steps describes in the process are implemented mainly using these algorithms.

1. WaterShed Algorithm :

Different approaches may be employed to use the watershed principle for image segmentation.

* Local minima of the gradient of the image may be chosen as markers, in this case an over-segmentation is produced and a second step involves region merging.
* Marker based watershed transformation make use of specific marker positions which have been either explicitly defined by the user or determined automatically with morphological operators or other ways.

1. Level Set Algorithm:

The central idea is to represent the evolving contour using a signed function, where its zero level corresponds to the actual contour. Then, according to the motion equation of the contour, one can easily derive a similar flow for the implicit surface that when applied to the zero-level will reflect the propagation of the contour. The level set method encodes numerous advantages: it is implicit, parameter free, provides a direct way to estimate the geometric properties of the evolving structure, can change the topology and is intrinsic.

**Implementation**

The Implementation is a proper sequential Implementation. All the steps described are implemented one by one.

1st Step : **Filtering the image** :

In this some minimum and maximum Fraction Digits are set. Based on that Hybrid2dMedianizer is called.

Imp = Filter(imp);

After that minimum and maximum Fraction Digits are reset.

Imp\_filter2 = Hybrid2dMedianizer(imp\_filter, nsize);

2nd Step: **Converting Image to Binary image.**

This is simply done by running inbuit plugin named “Binary”.

IJ.run(“Make Binary”,null);

3rd Step : **Applying watershed Algorithm**

The above watershed algorithms is simply implemented by calling the functions

“watershed”.

IJ.run(imp,“Watershed”,null);

4th Step **: Applying Level Set Segmentation**.

The above described Lever set Segmentation is also implemented by the code below.

IJ.run(“Level Set”,null);

The other important implementations are as below :

GroupedZProjector Z2= **new** GroupedZProjector();

Z2.run("");

Thresholder T = **new** Thresholder();

T.run("");

ParticleAnalyzer pa = **new** ParticleAnalyzer(12, measurements, rt, 1200,

99999999);

pa.analyze(proimg\_);

In the Level Set algorithm the seed point we have used is the Centroid of the neurons which we obtain from the Particle analyzer.

**ScreenShots**

**Below are some of the screens of our project**

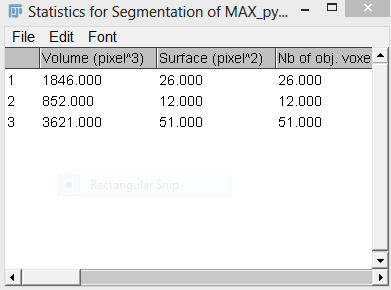
* **Z projection converted to binary:**

**C:\Users\Bhavani\Desktop\New folder\MAX_pyramidalcells.tif**

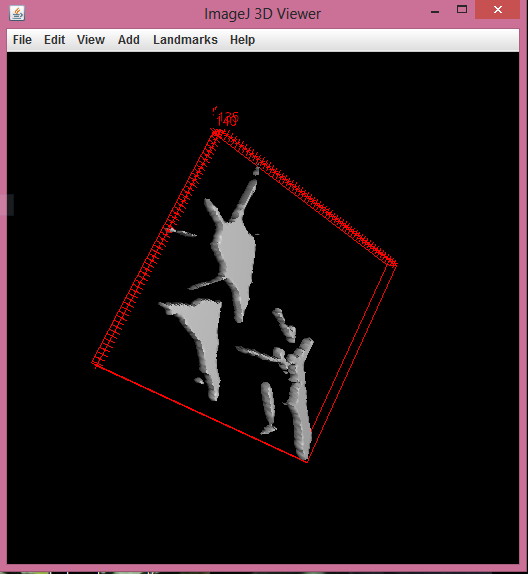
* **Sum of slices after segmentation:**

**C:\Users\Bhavani\Desktop\New folder\sum.tif**

* **Getting the volume and surface area of the neurons:**

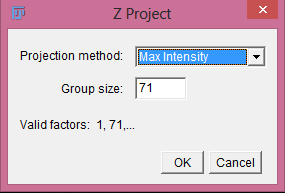
****

* **3D View of the segmented image:**

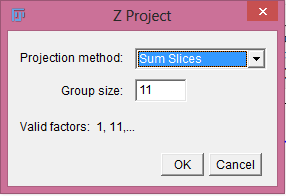


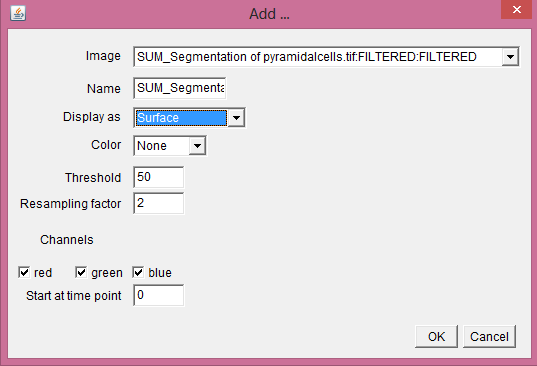
**Plugin Execution**

1. Run Fizi
2. Open Image
3. Run Plugin "**NEURON SEGMENTATION**"
4. Select Max\_Intensity from Z Project Window. This process continues and we will get Volume and Surface area statistics and centroid values.



1. There is no need to change any value, if any window appears. Just click "**OK**" to continue the process.
2. Finally we will get another Z project window to stack image slices. So, we have to select "**SUM SLICES**" in projection method, we processed only 11 slices because in these slices we can view the neurons clearly.



1. After the completion of above step we will get sum sliced image and the 3D image which is the other requirement of the project. For generating the 3D view we have to select **SURFACE** from "Display as" field

**Conclusion**

In this project we have successfully identified the Location of Neurons, their Surface are ,their Volume. We have also successfully applied 3D view for the segmented Neurons.

**References**

1. http://microscopy.berkeley.edu/courses/dib/sections/04IPIII/particles.html
2. http://en.wikipedia.org/wiki/Pixel\_connectivity
3. http://www.csee.usf.edu/~manohar/Papers/Segmentation/Seeded%20Region%20growing.pdf
4. http://www.orfeo-toolbox.org/SoftwareGuide/SoftwareGuidech16.html
5. http://textofvideo.nptel.iitm.ac.in/117105079/lec31.pdf
6. http://www.it.uu.se/edu/course/homepage/bild1/vt09/lectures/L08VT09.pdf