

MANUAL FOR SEGMENTATION WITH TomoSegMemTV

November 2013

Contact: an.martinez.s.sw@gmail.com

1	INTRODUCTION	1
2	INSTALLATION.....	2
3	USAGE.....	2
3.1	Example of usage	3
3.2	Help.....	4

1 INTRODUCTION

This is a manual for the package **TomoSegMemTV** that is designed for assisting in the segmentation of membranous structures in tomography. This package contains a set of MATLAB® functions. Nevertheless, the kernels of some computationally intensive operations (in *mex* directory) are binaries compiled for being used from Matlab, the code of these operations are not included in this package. The package **SynapSegTools** is an extension that contains graphic tools for advanced segmentation of cryo-tomograms of synaptosomes from the pre-processing carried out by TomoSegMemTV.

This package contains two main functions: **tomosegmentv** and **global_analysis**. The first one carries out a local analysis based on tensor voting algorithm and finds potential membrane-like centrelines while tries to fill up their structural gaps and discard structures adhered to them, in addition the

normal of the local surface at every point is also generated. The second one applies a global analysis based in the connectivity among structures and classifies them according to their volume. Both functions allow carrying out generic segmentations of membranes in tomography. The rest of functions are intermediate operations such as tensor voting, Gaussian filtering, derivatives, eigen-problem resolution, etc. Therefore, an advanced user could use them for further developments. An in depth description of the procedure for segmentation and the theoretical aspects of the code contained in this package is published in:

Martinez-Sanchez A., et al. Robust membrane detection based on tensor voting for electron tomography. sent to *J Struct Biol.* (2013)

2 INSTALLATION

Make the directory *source*, which contains the package, accessible from Matlab. Open Matlab, but before starting to use the package the directory *mex* must be added to Matlab's path. In Matlab Command Window:

```
>> addpath( 'path-to-tomosegmentv-package/source/mex/' );
```

Now the functions of the package are ready to be run in the Matlab environment from the Command Window.

Currently there exists versions of this package for MacOS 64 bits and Linux 64 bits. Contact with the developer (an.martinez.s.sw@gmail.com) if you have a different platform.

This package does not include functions for loading and saving tomograms from and to Matlab environment. This functionality can be acquired by installing TOM Toolbox (http://www.biochem.mpg.de/278655/tom_e), which allows loading and saving tomograms in standard formats such as MRC or EM.

3 USAGE

This package contains two main functions: **tomosegmentv** and **global_analysis**. The first one carries out a local analysis based on tensor voting algorithm and finds potential membrane-like centrelines while tries to fill their structural gaps and discard structures adhered to them, in addition the normal of the local surface at every point is also generated. The second one applies a global analysis based in the connectivity among structures and classifies them according to their volume. In Subsection 3.1 is available an example of usage for segmenting with TomoSegMemTV package.

The rest of functions are supporting operations such as tensor voting, Gaussian filtering, derivatives, eigen-problem resolution, ..., and advanced user could use

them for further developments. Subsection 3.2 shows how to access to a list with the interface of the functions contained in the package.

3.1 Example of usage

1) Download and extract (the file is compressed) the tomogram with ID 1155 from EMDataBank (<http://www.emdatabank.org/>).

2) Load the tomogram to Matlab environment. Despite of the file extension is .map it has MRC format. If TOM Toolbox is used then Matlab's commands are:

```
>> tom_mrcread(); % Select the tomogram file in the displayed window
>> T = single( ans.Value );
>> clear ans;
```

3) Tensor Voting based local filtering. Run **tomosegmentv** function in Matlab's command window. F is the filtered tomogram with the membrane centrelines pre-detected and Vi (optional) is a tomogram with the i-component of the local surface normal. The normals can be used for advanced post-processing, for example the package SynapSegTools uses them. In the Matlab's command window:

```
% sv=10 - Scale for Tensor Voting
% s=2 - Scale for the scale space
>> [F,Vx,Vy,Vz] = tomosegmentv( T, 10, 2 );
```

4) Global analysis of the pre-filtered tomogram based on connectivity where they are labelled according to their volume. In the Matlab's command window:

```
% tb=8 - Density threshold for binarizing
% c=18 - 3D connectivity used (6, 18 (default) and 26 available)
% v=1 - Verbose mode activated
>> C = global_analysis( F, 8, 18, 1 );
```

5) (Optional) Volume thresholding while volume labels are preserved. Directly this can be obtained through the next Matlab command:

```
% tb=6000 - Volume threshold
```

```
>> S = (C>6000) .* C;
```

Results inspection can be done directly through Matlab's environment (see Figure 1(A-C)), for instance the next command shows the slice Y=45 whereas the number of total slices in X and Z are 141 and 281 respectively:

```
>> figure(1), imshow( reshape(C(:,45,:),141,281), [] );
```

6) Store the segmented result in a standard format, in this example TOM Toolbox is used to generate a MRC file. Now the automatic segmentation performed can be opened from advanced tools for tomography visualization and edition such as Amira®. In the Matlab's command window:

```
>> tom_mrcwrite( S ); % Select the path and name of the output MRC file
```

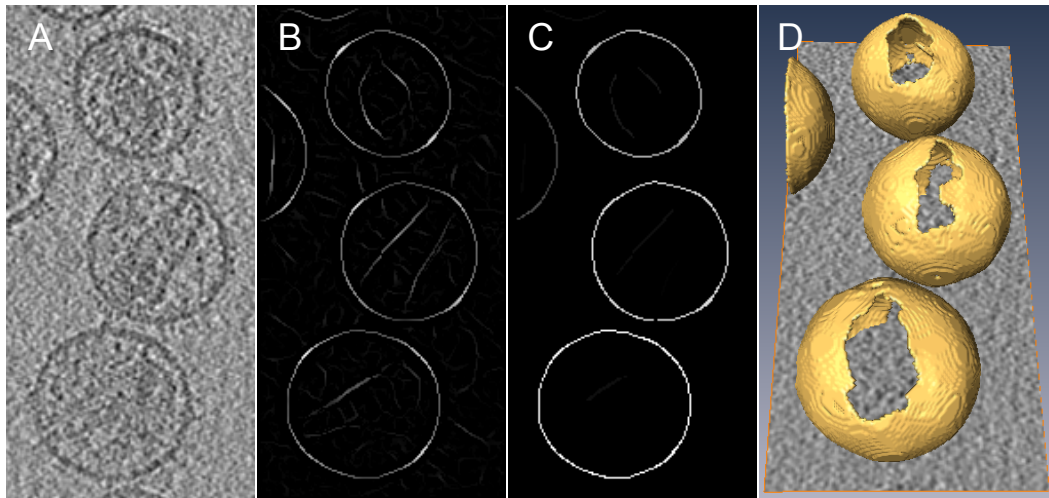


Figure 1. Example of cryo-tomogram segmentation with TomoSegMemTV. (A) Original tomogram downloaded from EMDataBank (ID: 1155). (B) Result of applying *tomosegmentv* (step 3). (C) Result of applying *global_analysis*, the gray level encodes the volume of each structure (step 4). (D) 3D Visualization with Amira® of the final segmentation.

3.2 Help

The file **help_report_matlab.html** attached with the package documentation contains a list and a brief description of the purpose and interface of the package functions. This information is also available for every function by using the command **help name-of-function** in Matlab.

This package is still under revision so any feedback is welcomed.