AnalyzeSkeleton

Analysis of 2D and 3D skeleton images.

Introduction

This free code is an [ImageJ](#) plugin that allows analyzing 2D or 3D skeleton images. AnalyzeSkeleton is also integrated in [Fiji](#).

Author

[Ignacio Arganda-Carreras](#)

General Description

For a detailed description of the plugin, please visit its Fiji website: http://fiji.sc/AnalyzeSkeleton

This plugin tags all pixel/voxels in a skeleton image and then **counts all its junctions, triple points and branches and measures their average and maximum length**. The tags are shown in a new window displaying every tag in a different color. See [Skeletonize3D](#) for an example of how to produce skeleton images.

The voxels are classified into three different categories depending on their 26 neighbors:

- **End-point** voxels: if they have less than 2 neighbors.
- **Junction** voxels: if they have more than 2 neighbors.
- **Slab** voxels: if they have exactly 2 neighbors.

Let's see an example of this classification:
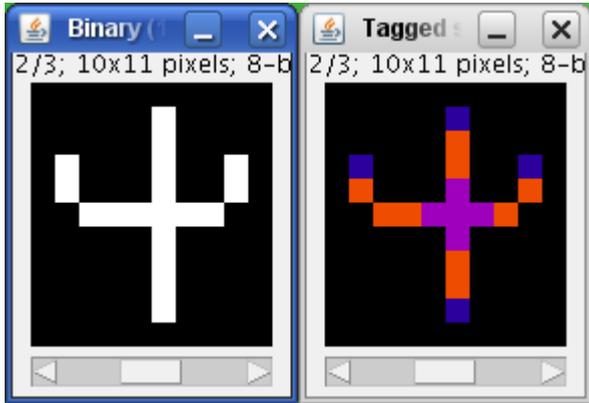
End-point voxels are displayed in **blue**, slab voxels in **orange** and junction voxels in **purple**.

After classification, a "Results" window is displayed showing the number of voxels of every type, branches, junctions and triple points (junctions with exactly 3 branches) and the average and maximum length of branches for every skeleton in the image. AnalyzeSkeleton is able to process up to $2^{15}-1$ (Short.MAX_VALUE) skeletons in one single image.

When calling the plugin, if the "Show detailed information" checkbox is marked, a complementary results table called "Branch information" is shown.

In this table we display all branches information:

- skeleton ID,
- calibrated branch length,
- 3D coordinates of the extremes of the branch (the so-called V1 and V2 vertices),
- and the Euclidean distance between those extreme points. This value has proven to be a good indicator of the tortuosity of the 3D object.

The branches are sorted by decreasing length.

Notice here that the number of junction voxels is different from the number of junctions since some junction voxels can be neighbors of each other.

Installation

Download AnalyzeSkeleton_-2.0.0-SNAPSHOT.jar in your ImageJ plugins folder and then restart ImageJ or simply apply the command *Help>Update Menus*. Then, you can call the plugin at *Plugins>Skeleton>Analyze Skeleton (2D/3D)*.

Note that if you download the jar file into a plugins sub-folder, then no *Skeleton* sub-menu will be created and the *Analyze Skeleton (2D/3D)* command will appear directly in the corresponding sub-folder menu.

Download

The latest release jar containing the class files can be downloaded from here:

https://github.com/fiji/AnalyzeSkeleton/releases

The source code can be found on GitHub.

References - How to cite

If you need to cite the plugin, please do so by citing the following paper:

- Ignacio Arganda-Carreras, Rodrigo Fernandez-Gonzalez, Arrate Munoz-Barrutia, Carlos Ortiz-De-Solorzano, "3D reconstruction of histological sections: Application to mammary gland tissue", Microscopy Research and Technique, Volume 73, Issue 11, pages 1019–1029, October 2010.

The shortest path calculation and its applications have been published as:

- G. Polder, H.L.E Hovens and A.J Zweers, Measuring shoot length of submerged aquatic plants using graph analysis (2010), In: Proceedings of the ImageJ User and Developer Conference, Centre de Recherche Public Henri Tudor, Luxembourg, 27-29 October, pp 172-177.

AnalyzeSkeleton makes also part of BoneJ, a plugin for bone image analysis in ImageJ:

- Michael Doube, Michal M. Klosowski, Ignacio Arganda-Carreras, Fabrice P. Cordelieres, Robert P. Dougherty, Jonathan S. Jackson, Benjamin Schmid, John R. Hutchinson, Sandra J. Shefelbine, BoneJ: Free and extensible bone image analysis in ImageJ, Bone, Volume 47, Issue 6, December 2010, Pages 1076-1079.

License

This program is **free software**; you can redistribute it and/or modify it under the terms of the **GNU General Public License** as published by the Free Software Foundation (http://www.gnu.org/licenses/gpl.txt).

This program is distributed in the hope that it will be useful, but WITHOUT ANY WARRANTY; without even the implied warranty of MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the GNU General Public License for more details.

Changelog

All changes can be seen in the GitHub [source repository](#).

2010/01/12: Thanks to Peter Marks, fixed bug to visit properly the trees when starting in junctions (they were not being added to the revisit list, what some times made some trees to be split).

2009/12/04: Added quadruple point calculation.

2009/09/13: Removed Log window and created Detailed information option and Branch information table.

2009/09/02: Added capability to detect and prune skeleton cycles.

2009/08/10: Fixed small bug to treat the special case of one single circular skeleton.

2009/08/06: Fixed 2 bugs: calculation of branches between junctions and number of slab voxels in a circular tree.

2009/08/05: Fixed bugs that slowed down the calculation of the number of actual junctions.

2009/06/19: Fixed bugs in some calculations, increased the number of skeletons from 255 to 2^1?-1, and added new (previously untreated) cases.

2009/04/07: Added different calculations for every skeleton in the image and some extra information in the log window.

2009/03/05: Added maximum branch length calculation.

2008/11/19: Added triple points calculation and fixed bug to avoid modifying the original image and preserve its calibration in the output image.

2008/11/16: First release.

Known Bugs

Since the 2009/09/02 version of the code, the possible cycles or loops in the skeleton can be detected and pruned previous to the analysis. The cycle detection is based on a classical Depth-First Search (DFS) in the skeleton. The only known limitation of this approach is shown in the presence of nested loops. In those cases, a second call to the plugin is usually enough to eliminate all the

remaining loops.