Contents

List of Figures

List of Tables

A. Guide on how to use NEFI

A.1. Properties of ideal and non-ideal images

In order to obtain good graph extraction results, **NEFI** relies on input of certain quality. In this section we give a description of desirable properties of ideal input images.

Since, the determining factor of the quality of **NEFI's** graph extraction is the segmentation step, ideal images should enable a nearly perfect separation of foreground and background. Such images should have a high contrast between the depicted structures of interest and the background. At the same time it is very important that images are free of strong reflections or shadows because such areas are likely to show an even higher contrast to the background than the actual structures of interest. As a result, the segmentation algorithms are prone to identify these regions as foreground causing the actual structures of interest to be ignored. The presence of strong color or brightness gradients can have similar detrimental effects and should thus be prevented if possible. See Figure A.1 and Figure A.2 for examples of challenging images which **NEFI** will have difficulties working with.

Fix the typesetting of the images in this chapter ...

Another factor that influences segmentation, and by extension graph detection, are contaminations of different origin. Parts of the image might not belong to the object of interest at all. Such regions should be removed before loading the image into **NEFI**, see Figure A.3.

Figure A.4 depicts an image lacking in a similar way. The image contains objects that are technically part of the network one would be interested to extract, however, they are not suited very well to be represented as a graph. In particular, they will be picked up correctly in the segmentation step yielding four large areas of white pixels. Subsequently, thinning will try to reduce these to a lines resulting in more or less unpredictable results. While the remaining parts of the structure will be processed correctly, such images do not constitute ideal candidates for processing with **NEFI**.

Summarizing this information we can collect the following desirable properties for an ideal image:

- High contrast between foreground and background.

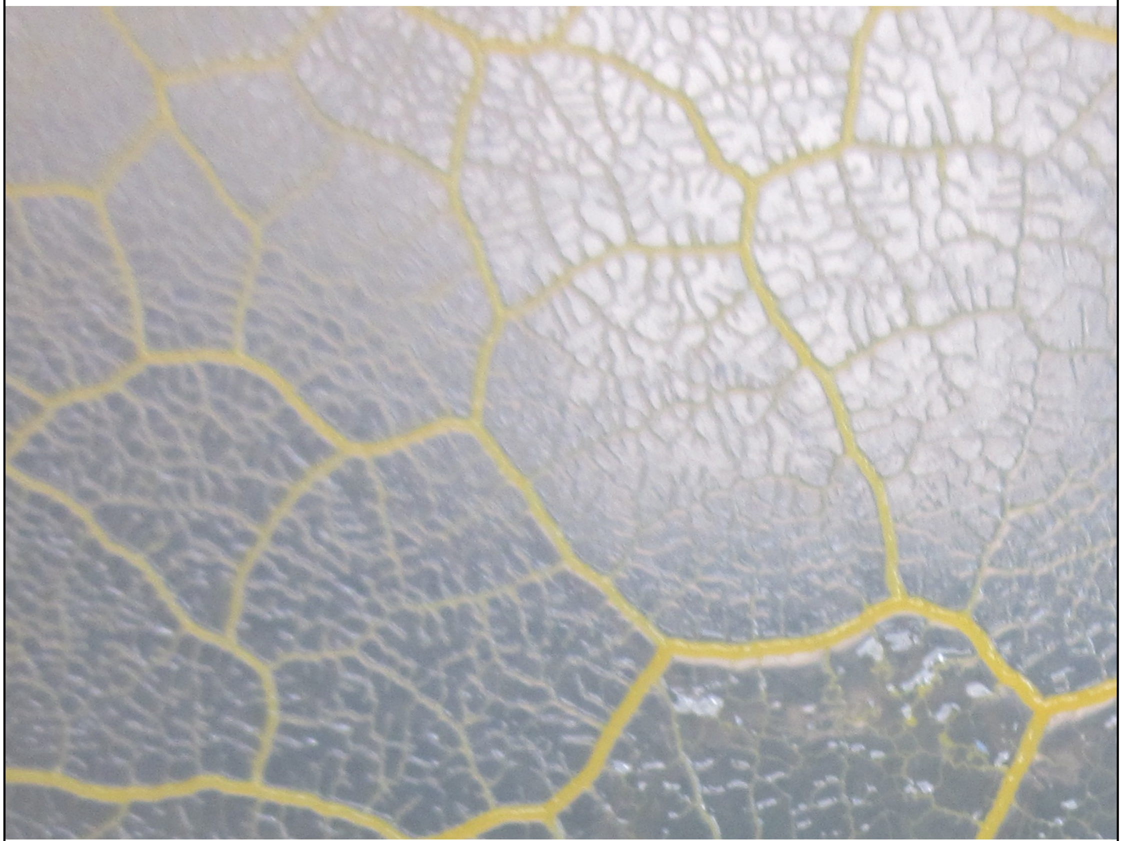

Figure A.1.: This image of *P. polycephalum* contains strong light reflections in the upper right quadrant which will cause **NEFI's** segmentation algorithms to be lead astray. The fact that the image is not properly focused is less of a problem in comparison.

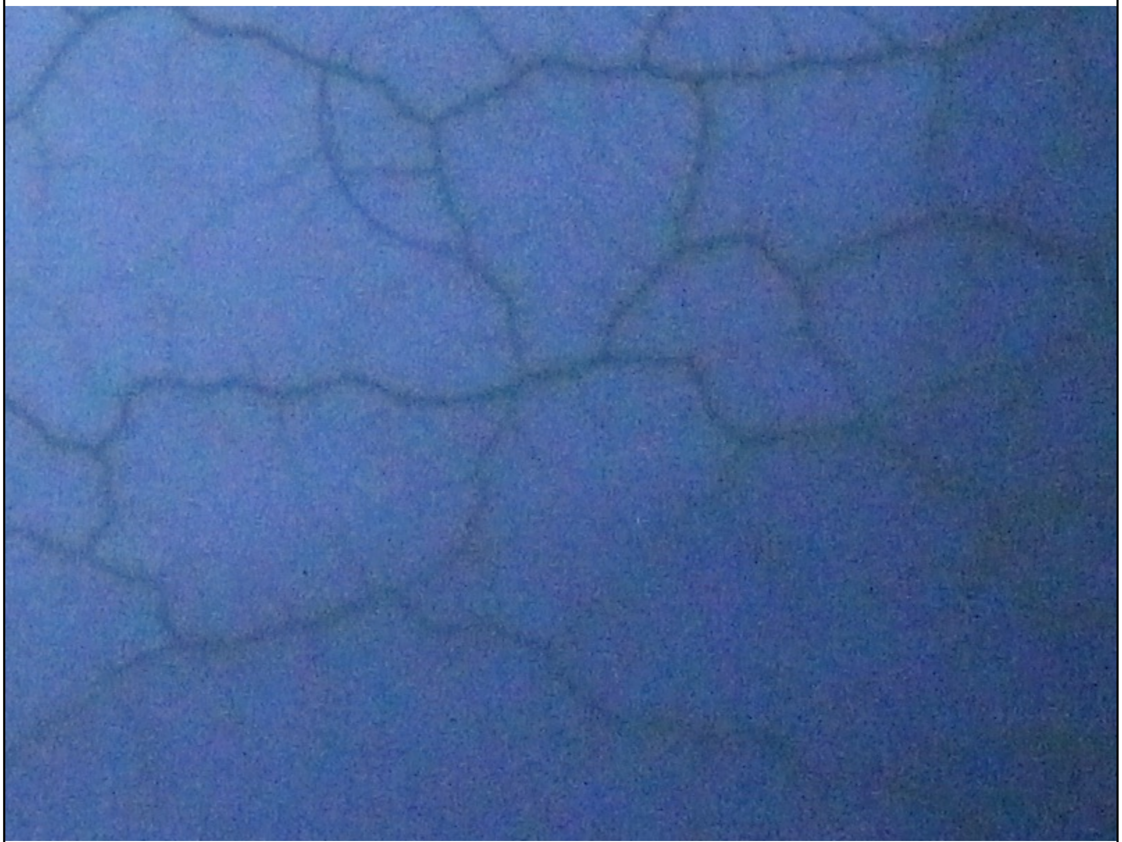

Figure A.2.: This image of *P. polycephalum* was not illuminated evenly from below. As a result it contains a brightness gradient which is detrimental for many of the segmentation algorithms currently implemented in **NEFI**. The fact that the image contains a lot of visible noise makes it a bad candidate for processing with **NEFI**.

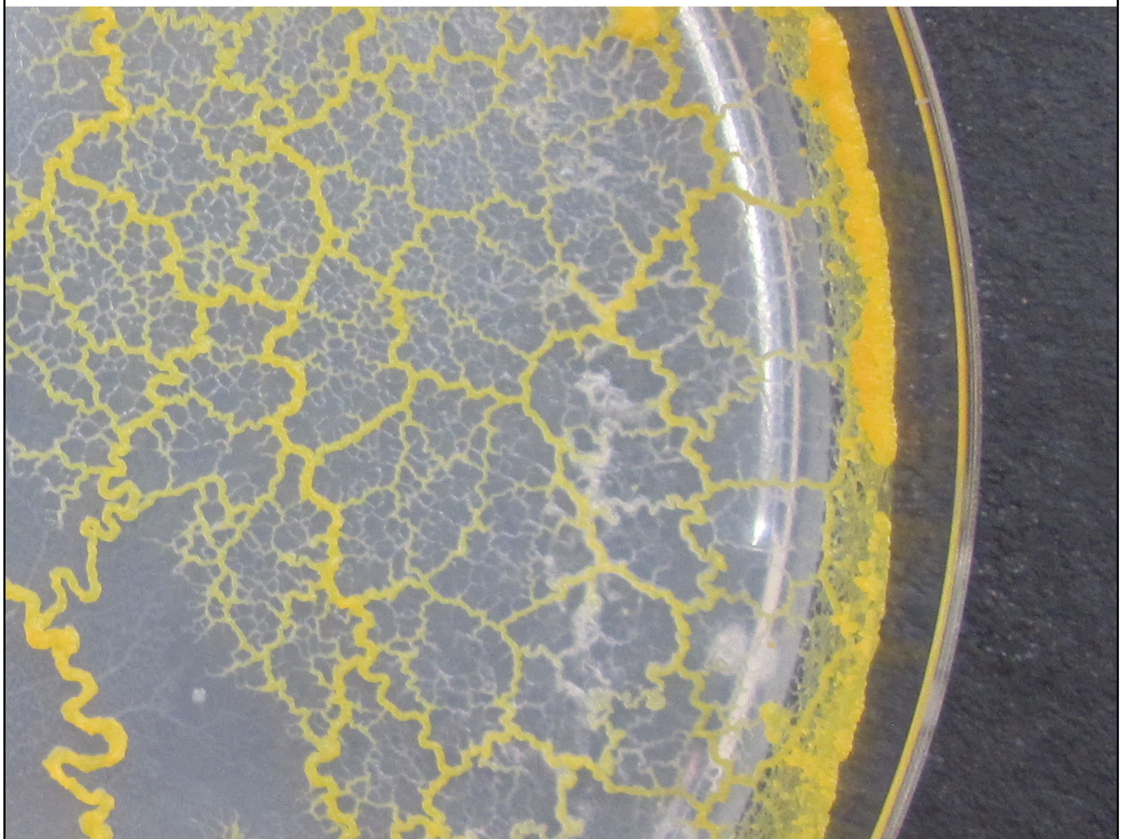

Figure A.3.: In addition to the network of *P. polycephalum* the image contains the edge of a petri dish and pieces of the background on the right hand side (as well as light reflections). Prior to any attempt of processing the image, petri dish and background should be removed.

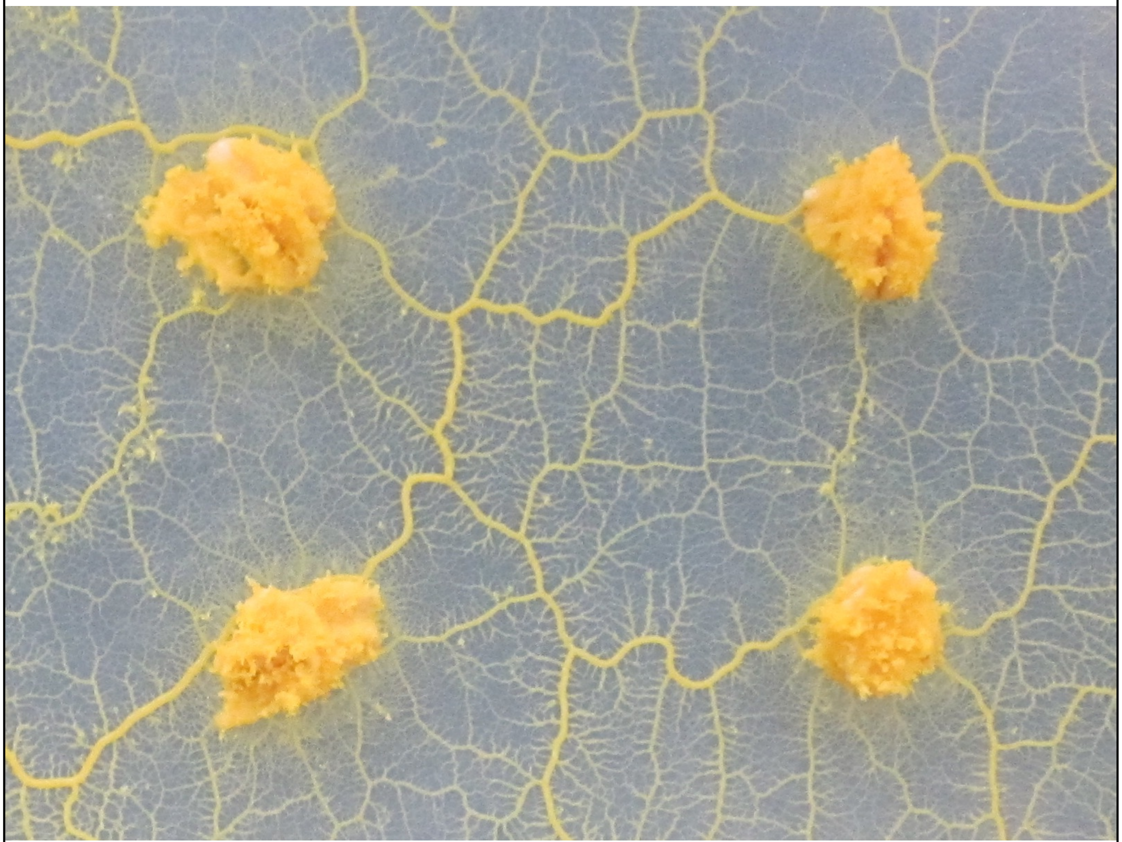

Figure A.4.: The network of *P. polycephalum* depicted in the image contains 4 massive, non-network-like regions (oat flakes completely covered by the mold). After segmentation, thinning will try to reduce these regions to lines leading to unpredictable results. The remaining parts of the network, however, will be extracted correctly.

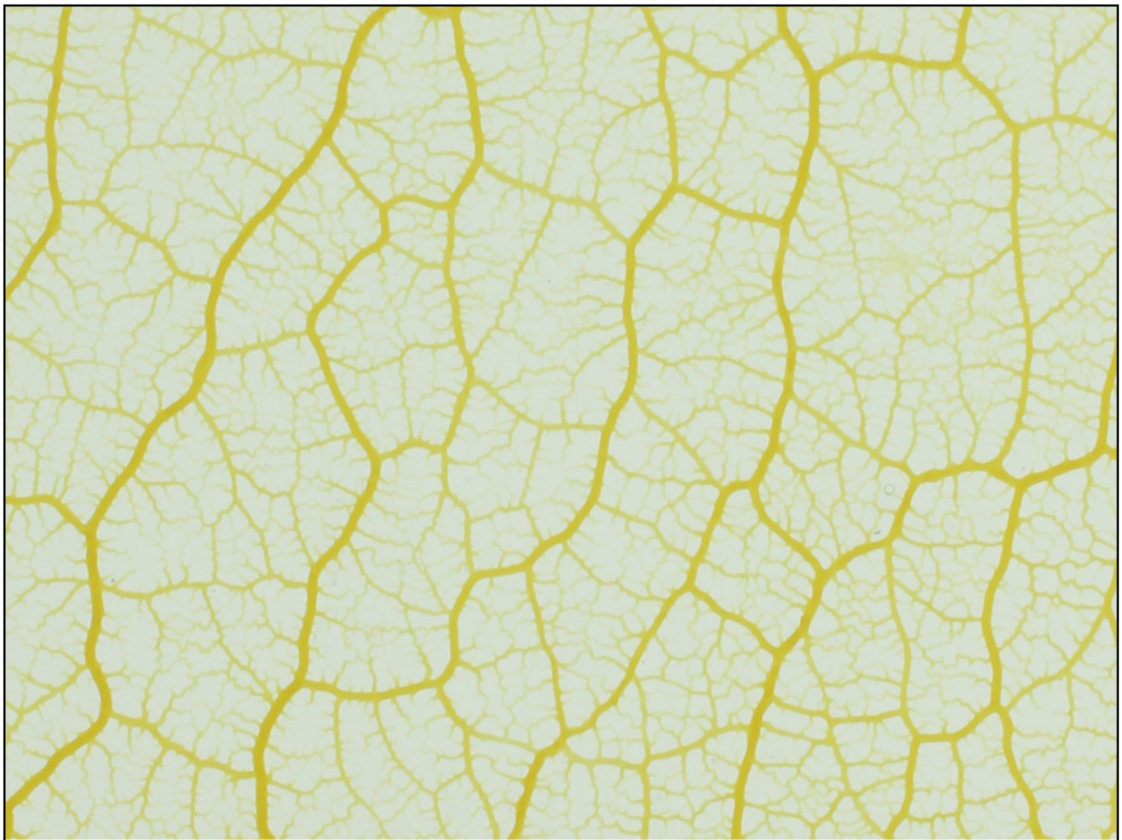

Figure A.5.: An ideal image of *P. polycephalum*. The image has been obtained using bright field illumination and was produced in a collaboration with the KIST Europe.

- Uniform background void of reflections, shadows as well as color or brightness gradients.
- No contaminations that might disrupt the segmentation process.

When producing images to be processed with **NEFI** one should strive to fulfill these properties whenever possible. See Figure A.5 and Figure A.6 for examples of very good input images.

A.2. How to deal with challenging images

fix the sizes of the images while youre at it ...

NEFI operates best on images produced under controlled laboratory conditions that fulfill the properties described in the last section. However, more difficult

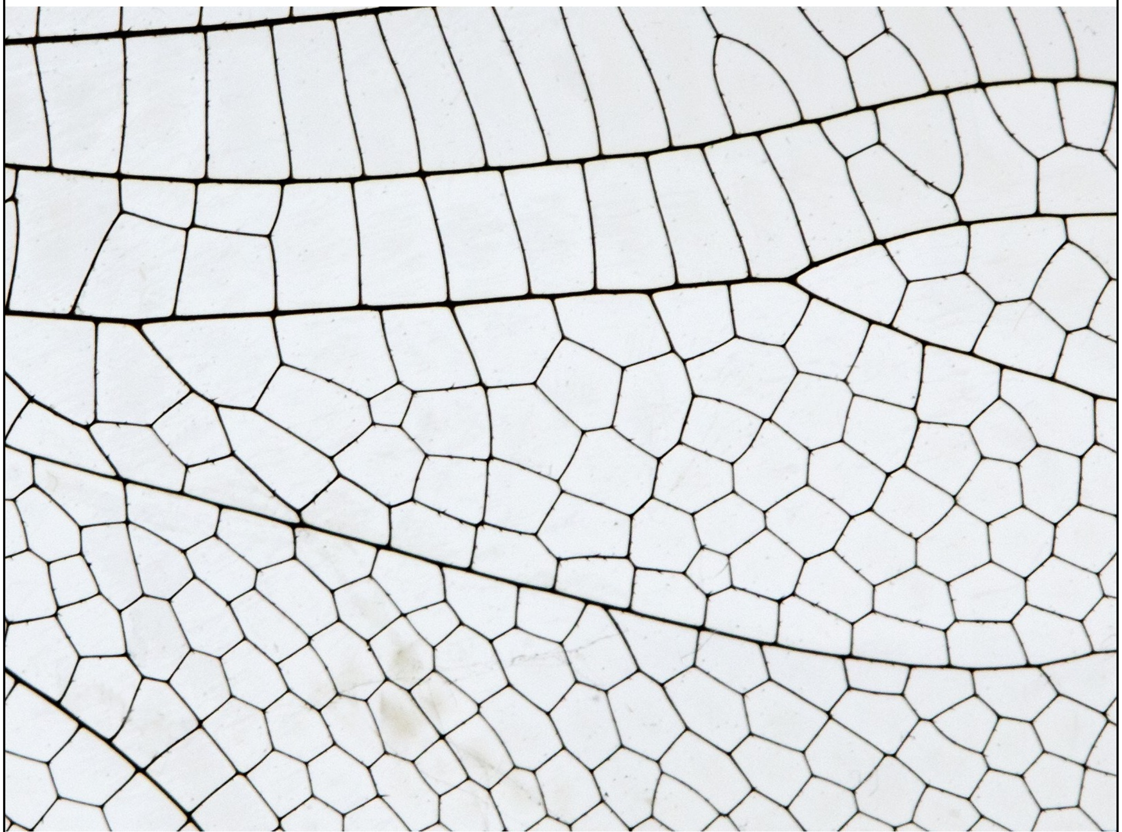

Figure A.6.: An ideal image depicting a detail of the wing of *A. junius*. Image courtesy of Pam and Richard Winegar.

inputs may still be processed, but most likely at the cost of reduced quality. Based on our experience when dealing with more challenging input and the results of our evaluation, we are able to formulate the following recommendations for the usage of **NEFI**:

Otsu's method may be used on noisy or blurred images. It will do reasonably well as long as if the image has a high contrast between network and background. Adaptive threshold, watershed based on adaptive threshold, and GrabCut with deletion and erosion may even perform slightly better under these conditions. Otsu's has the advantage that no parameters have to be set in order to get good results. The choice should be based on the desired degree of resolution of the extracted graph. We would recommend these methods to process Figure A.3 after clipping the image accordingly.

Adaptive threshold and watershed based on adaptive threshold may be used if differences in contrast between foreground and background are local and not too strong. If this is the case good results may still be obtained. Both methods allow to analyze images that contain a color gradient in the background or that contain edges which have differing levels in brightness. One might try to process images like Figure A.1 and Figure A.2 with these methods while experimenting with different parameter settings. However, such images remain challenging.**NEFI's** current algorithms may not deliver sufficient results.

Preprocessing methods like Gaussian and Median Blurring, Denoising as well as Bilateral Filtering can be used to remove small artifacts, contaminations or irregularities from the image. Although these methods can reduce the amount of artifacts produced during segmentation and thinning, improvements come at a price. For example, imposing a strong blur causes the depicted edges to appear slightly wider. This effect will propagate through the pipeline causing the edge widths of the final graph to overestimate the true widths depicted in the image. We recommend to use preprocessing with care especially if a high accuracy regarding edge weights is required.

Graph filtering enables the removal of unwanted artifacts and spurious vertices caused by irregularities in the input which propagate through the pipeline. However, filtering can only repair the result up to a given point. While the ability to add custom filters is powerful, it appears pointless to filter a graph established on the basis of a failed segmentation. We encourage users to visually verify the integrity of the established graph and only then to proceed with filtering.

We anticipate users to encounter images **NEFI's** algorithms will be unable to handle properly. In this situation the user will have to rely on different segmentation solutions.

As a first suggestion, we recommend to user to look for available software specializing in segmentation. In this regard we like to point out the Kitware, ITK Project [2]. It's *C++* code base has been developed by the medical image processing community and serves as the basis for many other specialized tools that deal with image processing as well as segmentation.

After a segmented image has been obtain using specialized third party segmentation software, NEFI can take this image and proceed with graph detection and filtering directly.

If no proper software is available, the user has the option to extend **NEFI's** segmentation capabilities by adding more sophisticated code. When doing so, one can built on top of existing features already implemented in **NEFI**. The literature offers a wealth of different approaches to segmentation leading to algorithms of varying complexity. Before diving into any implementation efforts, we strongly recommend to survey existing methods and their domains of effectiveness by consulting [1], [3], [4].

Bibliography

[1] R. Chellappa and B. Manjunath, "Texture classification and segmentation: tribulations, triumphs and tributes", English, in *Foundations of Image Understanding*, ser. The Springer International Series in Engineering and Computer Science, L. Davis, Ed., vol. 628, Springer US, 2001, pp. 219–240, ISBN: 978-1-4613-5599-1. DOI: 10.1007/978-1-4615-1529-6_8. [Online]. Available: http://dx.doi.org/10.1007/978-1-4615-1529-6_8.

[2] H. J. Johnson, M. McCormick, L. Ibáñez, and T. I. S. Consortium, *The ITK Software Guide*, Third, *In press*, Kitware, Inc., 2013. [Online]. Available: %5Chref%7Bhttp://www.itk.org/ItkSoftwareGuide.pdf%7D%7Bhttp://www.itk.org/ItkSoftwareGuide.pdf%7D.

[3] N. R. Pal and S. K. Pal, "A review on image segmentation techniques", *Pattern Recognition*, vol. 26, no. 9, pp. 1277–1294, 1993, ISSN: 0031-3203. DOI: http://dx.doi.org/10.1016/0031-3203(93)90135-J. [Online]. Available: http://www.sciencedirect.com/science/article/pii/003132039390135J.

[4] D. L. Pham, C. Xu, and J. L. Prince, "Current methods in medical image segmentation 1", *Annual review of biomedical engineering*, vol. 2, no. 1, pp. 315–337, 2000.

Todo list