

## Graph-Based Neural Reconstruction from Skeletonized 3D Networks

We would like to thank the reviewers for their time.

### Reviewer 1

**A1.** We thank the reviewer for linking to these four recent papers, although they are working on a different issue. Our algorithm works by correcting the merge errors generated by current segmentation algorithms. The Funke et al. paper is actually a pre-paper arxiv submission for the CVPR conference. We take issue with the statement that “the paper does not adequately reflect prior art, rendering its own conclusions questionable.” In fact, reviewer 3 wrote as a strength “the list of references is impressive, demonstrating comprehensive knowledge of both recent methods and the application domain”.

**A2.** This claim refers to building on-top of existing segmentation pipelines. We will clarify this point in the next revision. LASH extracts a graph from superpixels generated from a watershed. We extract a graph from this output at a different abstraction layer.

**A3.** We don’t agree with the assertion that these are “drastic” examples but rather honest mistakes. We can easily add an additional reference for U-Net that extends the original paper to 3D. In fact, citing reference [18] for the multicut algorithm is not a mistake at all. We use the graph optimization library from Bjoern Andres<sup>1</sup> which cites [18] (his own work) for the multicut package. We will update the reference to Funke et al. as this was a simple LaTeX citation mistake.

**B1.** We agree that skeletonization can improve but our goal for this paper is to use an existing method and explore its application to new ideas.

**B2.** This is incorrect, the skeletons from the figures are from the Kasthuri et al. data.

**B3.** This is addressed in the paper. First the cubic region of interest is in nanometers, not voxels. From section 4.8, “Since our CNN only takes as input a region of the label volume we can train on anisotropic data and test on isotropic data.” In the same section we show the results when training on isotropic data (Fig. 9).

**B4.** We address this in the supplemental material. We can include a reference to the supplemental material in the future.

**B5.** Without the images, we avoid retraining for every new dataset that has different staining or resolution, which is every dataset that we receive from our biology collaborators. We will clarify this point in a revision.

**B6.** This analysis is provided in Sec. 4.7 and Table 2.

<sup>1</sup><http://www.andres.sc/graph.html>

**B7.** We are not using “lifted” multicut but only the greedy-edge contraction method.

**B8.** Section 3.4 states that acyclic partitionings are enforced. Since acyclic regions are not an inherent outcome of the multicut algorithm, this requires a modified multicut, but the modification is unspecified.

**B9.** Here we reference the possibility for additional constraints using our new novel framework. Currently that enforces neurons to be acyclic but in the future can use other information. The important idea was that this framework allows domain-specific constraints.

**C1.** We use NeuroProof because it scales to terabyte datasets. Flood filling networks provide better accuracy but are 25x slower and work on a single neuron at a time. For dense reconstruction, NeuroProof is state-of-the-art. Variation of Information overcomes previous limitations of rand error and is more widely used by the community [1, 2]. The SNEMI3D dataset is considered outdated as concluded in [1].

**C2.** Most of the current state-of-the-art methods are too slow to run on very large datasets.

### Reviewer 2

Unfortunately the challenge datasets (SNEMI3D, CREMI) are too small for meaningful improvement. The datasets presented in our paper are roughly ten times the size of these challenge datasets.

### Reviewer 3

#### Address acyclic nature of graph.

There are some polynomial time algorithms for finding the general minimum spanning forest on a general graph ( $O(V^2E)$ ). The multicut algorithm we use has a running time of  $O(V^2 \log(V))$ , although provides an approximation to the optimal solution. Exploring both of these algorithms is reasonable in a future revision.

### References

- [1] K. Lee, J. Zung, P. Li, V. Jain, and H. S. Seung. Superhuman accuracy on the snemi3d connectomics challenge. *arXiv preprint arXiv:1706.00120*, 2017. 1
- [2] J. Nunez-Iglesias, R. Kennedy, T. Parag, J. Shi, and D. B. Chklovskii. Machine learning of hierarchical clustering to segment 2d and 3d images. *PLoS one*, 8(8):e71715, 2013. 1

054  
055  
056  
057  
058  
059  
060  
061  
062  
063  
064  
065  
066  
067  
068  
069  
070  
071  
072  
073  
074  
075  
076  
077  
078  
079  
080  
081  
082  
083  
084  
085  
086  
087  
088  
089  
090  
091  
092  
093  
094  
095  
096  
097  
098  
099  
100  
101  
102  
103  
104  
105  
106  
107