1226
doi:10.1017/S1431927622005104
Microsc. Microanal. 28 (Suppl 1), 2022
© Microscopy Society of America 2022

Phantoms Improve Robustness of Deep Learning Automated Segmentation in Cryotomography

Jessica Heebner[1], Carson Purnell[1], Mike Marsh[2], Matthew Swulius[1*]
[1.] Pennsylvania State University College of Medicine. Department of Biochemistry and Molecular Biology Hershey, PA, USA.
[2.] Object Research Systems. Denver, CO, USA.
* Corresponding Author: mts286@psu.edu

Cryo-electron tomography (cryoET) provides the highest resolution structural biology findings on cells in a hydrated state, but thorough interpretation of these large noisy image volumes is severely constrained by image segmentation shortcomings. Image segmentation approaches are often painstaking and laborious and remain largely unautomated. We posit that this is the primary reason that many tomograms languish, unanalyzed, never being explored to their scientific potential. Given that a single microscope can now collect 100 tomograms per day, and this volume of underutilized data is projected only to grow [1–2], the structural biology community needs better methods to rapidly parse the molecular contents of cryoET data.

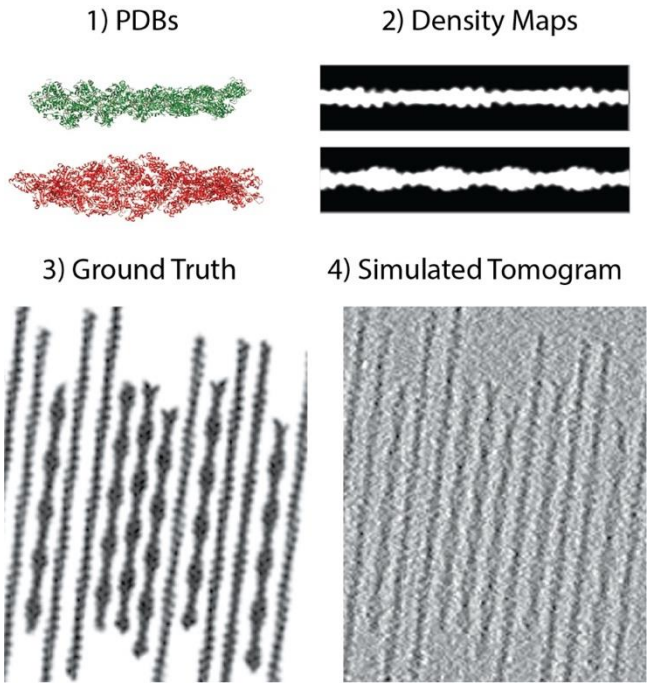

Figure 1. Simulation outline: 1) Obtain appropriate PDB structures. 2) Generate Density Map. 3) Create Ground Truth. 4) Simulated tomogram is reconstructed after missing wedge generation, noise addition and CTF convolution.

Deep learning is succeeding in many scientific imaging applications for analysis of large and complex datasets [3–8], and early successes in cryoET applications are demonstrated by work such as TomoSeg [9], DeepFinder [10], Cryo-CARE [11], HAL[12], and Isonet [13]. Luckily, sophisticated software interface and design has advanced to the point where non-experts can effectively train models spanning a variety of deep learning architectures, extending the platform to a wide audience with a diverse set of macromolecular interests.

Given the vast complexity of the biological proteome, it is necessary that we adapt our deep learning models to segment structures in a variety of contexts. Training models is easy to automate and upscale when training data are abundant. Drawing on the voluminous data archived in protein structure databases from around the world (PDB, EMDB, AlphaFold, etc.) offers an alternative to relying on empirical cryoET data for those trainings. Here we describe TomoSIM, our MATLAB-based software for producing simulated tomograms (phantoms), and we show how they have successfully advanced the robustness of our deep learning segmentation models.

https://doi.org/10.1017/S1431927622005104 Published online by Cambridge University Press

The TomoSIM toolchain starts with importing a model structure and calculating its density map, at a pixel size and resolution matching experimental conditions. Multiple instances of this map (sampling rotational space) are positioned within a virtual 3D volume. TomoSIM scripts then array globular proteins on a three-dimensional grid or place filamentous polymers (such as actin) within a stack of rotated sheets (Fig. 1), in order to better sample the effects of the missing wedge on image reconstruction. This simulated volume serves as a cytoplasmic phantom, and a simulated tilt series is produced with IMOD. Each tilt-projection is convolved with a computed contrast transfer function [14],

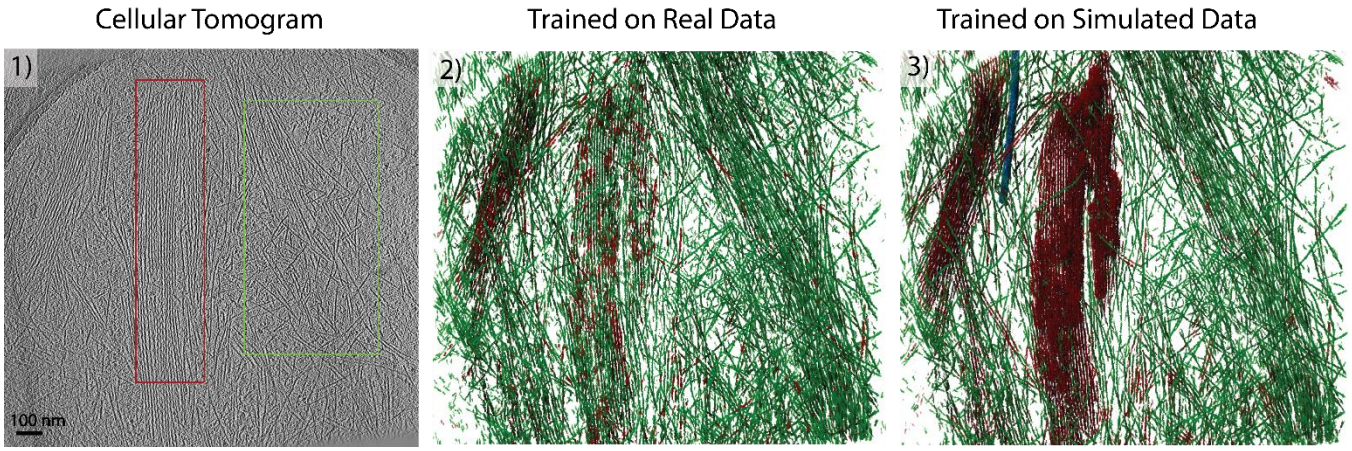

Figure 2. *Segmentation with and without simulation strategy.* 1) Tomogram to be segmented. Red box shows a cofilin-decorated actin rich region. Green box shows f-actin rich region. 2) 3D rendering of segmented data from network trained on the tomogram. Green class is f-actin. Red class is cofilin-decorated actin. 3) 3D rendering of segmented data from network trained with TomoSIM simulated data. Green class is f-actin. Red class is cofilin-decorated actin. Blue is a microtubule.

and various models of noise are added to simulate realistic tomographic conditions. Finally, the tilt series is reconstructed by weighted backprojection.

Image processing, deep learning training, image segmentation, and 3D rendering are performed with the Dragonfly software package [15]. Training data is generated by pairing the reconstructed phantom with labels that identify all of the pixels of the constituent macromolecular assemblies. Assigning pixels to their corresponding class is trivial, because of the constraints of how the phantoms were generated. Using the simulated tomogram as input data, and the segmentation as target output, a 5-slice U-Net (or other models) is trained in the Segmentation Wizard [16]. Within the wizard this network can be applied to real data and iteratively trained to improve the segmentation, using only a small volume of empirical data.

Large grids of mixed macromolecules can be rapidly generated, and creating training data is as simple as thresholding the reconstructed phantoms. Shown below, networks trained only on empirical data successfully identify almost all actin filaments, but fail to distinguish between actin filaments in different conformations (Fig. 2.2). When a network is trained using the perfectly segmented phantoms, however, the accuracy of the segmentation dramatically improves, and different conformations of actin can be reliably identified (Fig. 2.3). These results suggest a generalized protocol for training deep learning models to identify complex or fine-grained features within cryo-electron tomograms using

https://doi.org/10.1017/S1431927622005104 Published online by Cambridge University Press

independent, a priori training. TomoSim provides a platform for training any networks to high precision that can be tested against real world data rapidly [17].

References:

[1] G. Chreifi, S et al., *Journal of Structural Biology*, **205** (2019) 163–169. https://doi.org/10.1016/j.jsb.2018.12.008.
[2] G. Chreifi, S. Chen, & G. J. Jensen, *Journal of Structural Biology*, **213** (2021) 107716. https://doi.org/10.1016/j.jsb.2021.107716.
[3] S. M. Lee, J et al., *Journal of Thoracic Imaging*, **34** (2019) 75–85. https://doi.org/10.1097/RTI.0000000000000387.
[4] E. Moen et al., *Nat Methods*, **16** (2019) 1233–1246. https://doi.org/10.1038/s41592-019-0403-1.
[5] K. Wang, *Nature Methods*, **18** (2022) 454–462. https://doi.org/10.1038/s41592-021-01151-1.
[6] D. S. W. Ting et al., *Progress in Retinal and Eye Research*, **72** (2019). https://doi.org/10.1016/j.preteyeres.2019.04.003.
[7] M. D. Holbrook et al., *Tomography*, **7** (2021) 358–372. https://doi.org/10.3390/tomography7030032.
[8] J. Malimban et al., *Scientific Reports*, **12** (2022) 1822. https://doi.org/10.1038/s41598-022-05868-7.
[9] M. Chen et al., *Nature Methods*, **14** (2017) 983–985. https://doi.org/10.1038/nmeth.4405.
[10] E. Moebel, A et al., *Nature Methods*, **18** (2021) 1386–1394. https://doi.org/10.1038/s41592-021-01275-4.
[11] T. O. Buchholz, M. Jordan, G. Pigino, & F. Jug, *Proceedings - International Symposium on Biomedical Imaging*, **April** (2019) 502–506. https://doi.org/10.1109/ISBI.2019.8759519.
[12] X. Du, H et al., *Bioinformatics*, **37** (2021) 2340–2346. https://doi.org/10.1093/bioinformatics/btab123.
[13] Y.-T. Liu et al., *bioRxiv*, (2021). https://doi.org/10.1101/2021.07.17.452128.
[14] TOM Toolbox | Max Planck Institute of Biochemistry. https://www.biochem.mpg.de/6348566/tom_e
[15] Dragonfly 2021.3 (Computer Software). (2021) http://www.theobjects.com/dragonfly.
[16] B. Provencher, M. Ruslana, E. Yen, N. Piche, & M. Marsh, *Microscopy and Microanalysis*, **25** (2019) 1388–1389.
[17] J. Heebner and C. Purnell contributed equally to this research.

https://doi.org/10.1017/S1431927622005104 Published online by Cambridge University Press