Applying Deep Learning Algorithms To Track Cells In Time-Lapse Microscopy

Lianhao Zheng, Keara Lane
Department of Molecular Biosciences, Northwestern University, Evanston, IL, USA

Introduction

Salmonella enterica serovar Typhimurium (STm) is a gram-negative bacterium and a major cause of foodborne illnesses worldwide[1]. STm can be ingested, or phagocytosed, by macrophages, an immune cell type. Following phagocytosis, the host cell either degrades the bacteria or the bacteria replicate to high, medium, or low levels. However, the basis for these different STm fates remains unclear. One method to improve our understanding of the factors involved is to follow individual macrophages over the entire duration of infection. Live-cell microscopy or imaging is an ideal method for this, but it requires reliably tracking, or following, the cells over many hours. Current cell tracking algorithms can be computationally expensive and face challenges when used over a long time course. Through my eight-week research, I attempted to apply newer tracking algorithms that incorporate elements of deep learning, such as Deep Cell, Trackmate7, and Btrack to resolve these tracking issues[2–4]. Moreover, I optimized an analysis pipeline to track infected macrophages over time by integrating the tracking softwares to use on existing microscopy data.

Background

Tracking infected macrophages requires proper cell segmentation, or identification, and tracking.

Challenges with Tracking Cells:

- Irregular Cell Movements
- Requires correct cell lineage identification
- Cell Death
- Cell Entry and Exit
- Cell Division

Previous Methods:

- Thresholding (Segmentation)
- Manual Linear Sum Assignment Problem (LSAP)
- Nearest Neighbor approach
- Linked-based cell detection

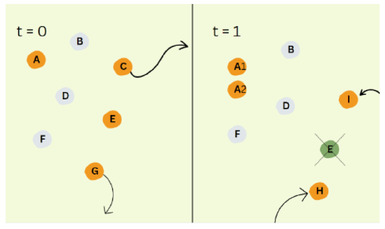


Figure 1. Cells across time (t=0, t=1) can perform cell division (A to A1, A2), cell death (E), cell entry (H, I), and cell exit (G, C).

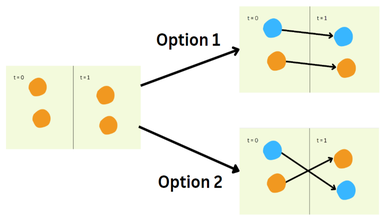
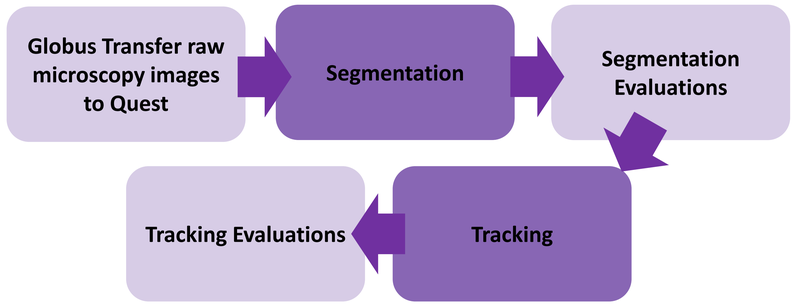


Figure 2. Multiple possible connections of cells or cell lineages between time points (t = 0, t = 1) are possible.

Newer Deep Learning (DL) Methods have been applied to segmentation and tracking. Segmentation examples include Cellpose and Stardist[5-6]. Tracking examples include Deep Cell, Trackmate7, and Btrack[2–4]. Notably, Trackmate7 uses a Labeled Image Detector to detect cells given segmented images and tracks detected objects using algorithms like the LAP tracker, which is based on the LSAP[4].

Methods/Results

Figure 3. General Pipeline Layout



The pipeline outlined above integrates segmentation along with tracking to automate image analysis of macrophages across time. 119 positions with 73 frames were segmented with *Cellpose* and *Stardist* on Quest, a High-Performance Computing cluster or collection of many separate computers that allows fast and parallel launching of jobs via BashScript[7].

Segmentation

Segmentation Measure (SEG), a measure based on intersection over Union, is shown below and used to determine performance of segmentation with Ground Truth data that I manually created on one frame with 15 positions [8-9].

$$J(S,R) = \frac{|R \cap S|}{|R \cup S|}$$

Cellpose was run with both the cyto2 and nuclei models as well as different Mask and Flow Thresholds. Cellpose resulted in a SEG score of 0.56. Stardist[5], which is based on the U-net Neutral network and star-convex polygons for localization of nuclei, performed significantly better with a SEG score of 0.78. These results contributed to using Stardist for segmentation.

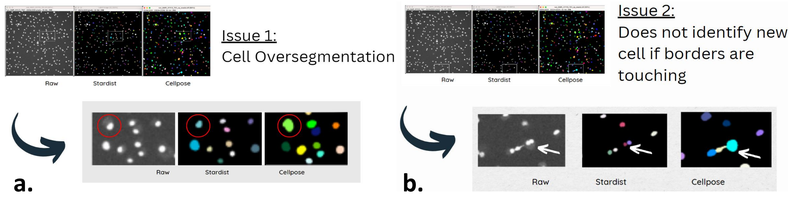


Figure 4. a-b, Segmentation issues associated with Cellpose.

Tracking

Tracking was only done with Trackmate7 and Deepcell due to time constraints.

DeepCell

DeepCell uses a DL model with a linear programming approach to track cells. A Python script with the pre-trained model was used to run Deepcell. Deepcell's issues were mainly misidentification of cells when excessive cell movement occurs across frames and after divisions, causing an excessive number of tracks.

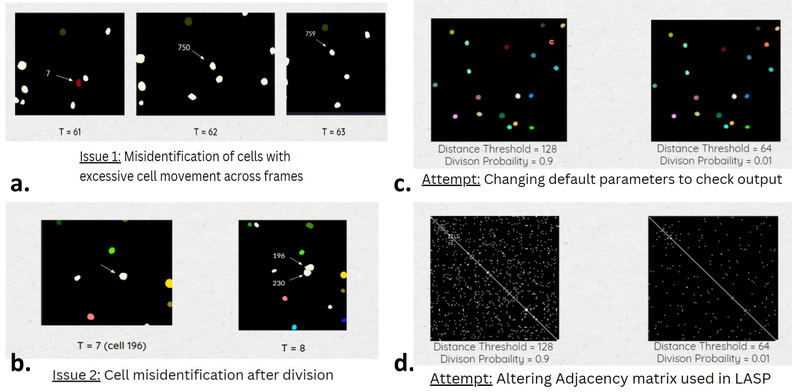


Figure 5. a-b, Issues with DeepCell Tracking. **c-d**, Parameters of pre-trained model (Distance Threshold, Division Probability) were altered to enhance tracking results but returned the same output. The adjacency matrix used in Deepcell was examined and proved to be affected. Thus, the pre-trained model isn't using the given parameters for tracking.

Trackmate7

Trackmate7's local tracking for each position of 73 frames took 1 minute using Fiji/ImageJ or a Jython script with the LAP tracker[10]. One issue is parent cells being recognized after divisions, but this can be corrected using its lineage detection tool—Trackscheme.

Ultimately, the final pipeline was based on Stardist and Trackmate7, consisting of a bash script for segmentation via Stardist on Quest and a Jython script for tracking with Trackmate7 locally.

Future Directions and Personal Statement

Evaluating the tracking via a quantitative measure: Evaluating the tracking results of the pipeline through the Tracking Score (TRA) will give a better representation of the accuracy of the overall pipeline[8-9].

Fixing and Optimizing the Trackers: DeepCell can be improved upon by allowing the default model to use the inputs provided. Trackmate7 can be improved upon by optimizing its feature penalties, gap closing, and segment splitting features.

Through this project, I gained experience working with unfamiliar programming languages such as Bash Script, Jython, running scripts on Quest, and the overall steps of deep-learning based segmentation and tracking algorithms.

Acknowledgments and References

Acknowledgments:

- Figures created in Canva
- Lane Lab : Madison S., Gloria A., Anika M., Dakshya K., Emily L., Emily F., Jonathan Z.,and Keara L.
- The study resulting in this publication was assisted by a grant from the Arch Scholars NU Bioscientist Program, which is administered by Northwestern University's Weinberg College of Arts and Sciences. However, the conclusions, opinions, and other statements in this publication are the author's and not necessarily those of the sponsoring institution.

References:

1. Sun, H. et. al (2020) "The Epidemiology of Monophasic Salmonella Typhimurium." *Foodborne Pathog. Dis.*
2. Moen, E. et al. (2019) "Accurate cell tracking and lineage construction in live-cell imaging experiments with deep learning." *bioRxiv* 803205
3. Ulicna, K., Vallardi, G., Charras, G. & Lowe, A. R. (2021) "Automated deep lineage tree analysis using a bayesian single cell tracking approach." *Front. Comput. Sci.*
4. Ershov, D. et al. (2022) "TrackMate 7: integrating state-of-the-art segmentation algorithms into tracking pipelines." *Nature Methods*
5. Schmidt, U., Weigert, M., Broaddus, C. & Myers, G. (2018) "Cell Detection with Star-Convex Polygons." *Springer International Publishing*
6. Stringer, C., Wang, T., Michaelos, M. & Pachitariu, M. (2021) "Cellpose: a generalist algorithm for cellular segmentation." *Nature Methods*
7. Quest High-Performance Computing Cluster: Information Technology - Northwestern University. it.northwestern.edu/departments/it-services-support/research/computing/quest
8. Maška, M. et al. (2014) "A benchmark for comparison of cell tracking algorithms." *Bioinformatics*
9. Matula, P. et al. (2015) "Cell Tracking Accuracy Measurement Based on Comparison of Acyclic Oriented Graphs." *PLoS One*
10. Schindelin, J. et al. (2012) "Fiji: an open-source platform for biological-image analysis." *Nature Methods*