

COMP9517: Computer Vision

2020 Term 2

Group Project Specification

Maximum Marks Achievable: 30

The group project is **worth 30% of the total course marks**.

Project work is in Weeks 6-10 with a demo and report due in Week 10.
Refer to the separate marking criteria for detailed information on marking.
Submission instructions and demo schedule will be released later.

Introduction

The goal of the group project is to work together with peers in a team of 4-5 students to solve a computer vision problem and present the solution in both oral and written form.

Each group must meet with their assigned tutor once per week in Weeks 6-9 and update the tutor on their progress. You may use the scheduled lab session hour on Fridays 1-2PM or arrange another time with your tutor. Tutors will set up a Team on Microsoft Teams for each group so that they can meet privately with each group. You may use the Team to communicate with your group members as well as your tutor.

Description

Tracking of biological cells in time-lapse microscopy images is one of the most common and important computer vision tasks in cell biology [1-3]. To study how cells move, divide, and interact under different conditions (healthy versus diseased), biologists often culture cells in a petri dish and then image them over time using a microscope. The resulting image sequences (videos) are usually too large and contain too many cells to track by hand.

Thus, computer vision methods are needed to automate the detection and tracking of the cells, as well as to perform subsequent quantitative analysis of cell motion. Many well-established computer vision methods in conjunction with machine learning methods are useful in these tasks. These may include methods for image preprocessing, feature extraction, classification, motion detection, tracking and recognition, using either unsupervised or supervised approaches, including various types of deep neural networks [4-9].

This project aims to familiarise students with some of the major methods involved in cell detection, tracking, and analysis, and develop their own solutions.

Tasks

The group project consists of three tasks described below, each of which needs to be completed as a group and will be evaluated for the whole group.

The image data to be used in the group project is taken from the international Cell Tracking Challenge (CTC) [10] and is provided as sequences of images (one sequence for each time-lapse microscopy recording). The data set contains multiple sequences from different biological experiments. The developed methods should work on all these data.

Task 1: Detect and Track Cells

Develop a Python program to detect and track all the cells within the image sequences. This means the program needs to perform the following steps:

- 1-1. Detect all the cells and draw a bounding box around each of them. For each image in a sequence the program should show the bounding boxes for that image only.
- 1-2. Draw the trajectory (also called track or path) of each cell. For each image in a sequence the program should show for each cell the past trajectory up to that time point.
- 1-3. Print (either as an output to the terminal or directly in the image window) the real-time count of the cells detected in each image of the sequence.

Task 2: Detect Cell Divisions

有丝分裂

Extend the program so that it can detect cell division (also called mitosis) events. For each dividing cell, the process of splitting of the mother cell into two daughter cells may take multiple time points to complete. The program should output the following:

- 2-1. Change the colour or shape (your choice) of the cell's bounding box for those time points during which the cell is in the process of dividing. After the division is complete, the program should track the two daughter cells as new cells.
- 2-2. Print (either as an output to the terminal or directly in the image window) the real-time count of the cells that are dividing at each time point.

Task 3: Analyse Cell Motion

Further extend the program so that it can analyse the motion of a selected cell. At any time point, the user should be able to select a cell, and the program should output the following (either to the terminal or directly in the image window):

- 3-1. Speed of the cell at that time point. This can be estimated by taking the Euclidean distance (in pixels) between the coordinates of the cell's bounding box center in the current time point and the previous time point, divided by the time difference (the latter is simply 1 frame, so the unit of speed is pixels/frame). Notice this means for the first time point of a cell's trajectory, no speed estimate can be computed.
- 3-2. Total distance travelled up to that time point. This is the sum of the Euclidean distances (in pixels) computed from the first time point of a cell's trajectory to the second, from the second to the third, and so on, until the current time point.
- 3-3. Net distance travelled up to that time point. This is the Euclidean distance (in pixels) directly between the cell's coordinates in the current time point and its coordinates in

- the first time point of its trajectory.
- 3-4. Confinement ratio of the cell motion. This is the ratio between the total distance travelled by the cell up to the current time point (computed in 3-2) and the net distance travelled up to the current time point (computed in 3-3).
-

Deliverables

The deliverables of the group project are 1) a group demo and 2) a group report. Both are due in Week 10. More detailed information on the two deliverables:

Demo: During the scheduled lecture and lab hours in Week 10, Thursday 6 August 2020 2-4PM and Friday 7 August 2020 12-2PM, group demos will be held. Each group will make a 10-minute online live demo to one tutor and one assessor, and students from other groups may tune in as well. The demo should include a short slide-show presentation (5 slides maximum) explaining your methods and evaluation, followed by a demonstration of your methods in action, and a brief discussion of how they perform on the given data. Afterwards, you will answer questions from the tutor/assessor/audience. All group members **must** be present for this demo. The demo roster will be released closer to the deadline.

Report: Each group will also submit a report (maximum 10 pages, [2-column IEEE format](#)) along with the source code(s), before 7 August 2020 23:59:59. The report should include:

1. Introduction: Discuss your understanding of the task specification and data sets.
 2. Literature Review: Review relevant techniques in literature, along with any necessary background to understand the techniques you selected.
 3. Methods: Justify and explain the selection of the techniques you implemented, using relevant references and theories where necessary.
 4. Experimental Setup: Explain the experimental setup and evaluation methods.
 5. Results and Discussion: Provide statistical and visual results, along with a discussion of method performance and outcomes of the experiments.
 6. Conclusion: Summarise what worked / did not work and recommend future work.
 7. Contribution of Group Members: State each group member's contribution in brief. In at most 3 lines per member, describe the component(s) each group member contributed to.
 8. References: List the literature references used in your work.
-

References

The following papers provide much useful information about microscopic image analysis and cell tracking. If the papers are not directly available (open access) by clicking the links, they should be available online via the UNSW Library.

- [1] E. Meijering, O. Dzyubachyk, I. Smal, W. A. van Cappellen. Tracking in cell and developmental biology. *Seminars in Cell and Developmental Biology*, vol. 20, no. 8, pp. 894-902, October 2009.
<https://doi.org/10.1016/j.semcdb.2009.07.004>
- [2] C.-M. Svensson et al. Untangling cell tracks: quantifying cell migration by time lapse image data analysis.

- Cytometry Part A, vol. 93, no. 3, pp. 357-370, March 2018. <https://doi.org/10.1002/cyto.a.23249>
- [3] A.-A. Liu et al. Mitosis detection in phase contrast microscopy image sequences of stem cell populations: a critical review. IEEE Transactions on Big Data, vol. 3, no. 4, pp. 443-457, October 2017. <https://doi.org/10.1109/TBDATA.2017.2721438>
 - [4] J. C. Caicedo et al. Evaluation of deep learning strategies for nucleus segmentation in fluorescence images. Cytometry Part A, vol. 95, no. 9, pp. 952-965, September 2019. <https://doi.org/10.1002/cyto.a.23863>
 - [5] T. Falk et al. U-Net: deep learning for cell counting, detection, and morphometry. Nature Methods, vol. 16, no. 1, pp. 67-70, January 2019. <https://doi.org/10.1038/s41592-018-0261-2>
 - [6] E. Moen et al. Deep learning for cellular image analysis. Nature Methods, vol. 16, no. 12, pp. 1233-1246, December 2019. <https://doi.org/10.1038/s41592-019-0403-1>
 - [7] Y. Li et al. Detection and tracking of overlapping cell nuclei for large scale mitosis analyses. BMC Bioinformatics, vol. 17, no. 1, p. 183, April 2016. <https://doi.org/10.1186/s12859-016-1030-9>
 - [8] X. Lou et al. Active structured learning for cell tracking: algorithm, framework, and usability. IEEE Transactions on Medical Imaging, vol. 33, no. 4, pp. 849-860, April 2014. <https://doi.org/10.1109/TMI.2013.2296937>
 - [9] E. Meijering et al. Methods for cell and particle tracking. Methods in Enzymology, vol. 504, no. 9, pp. 183-200, February 2012. <https://doi.org/10.1016/B978-0-12-391857-4.00009-4>
 - [10] V. Ulman et al. An objective comparison of cell-tracking algorithms. Nature Methods, vol. 14, no. 2, pp. 1141-1152, December 2017. <https://doi.org/10.1038/nmeth.4473>
-

Copyright: UNSW CSE COMP9517 Team

Released: 3 July 2020