

Cutting Constraints on Conservation Tracking

Project Report MLCV Summer 2017

Kodai Matsuoka
kodaig06@gmail.com

Yuyan Li
yuyan.li@gmx.net

Jui-Hung Yuan
j.yuan@stud.uni-heidelberg.de

Abstract

1. Introduction

To understand the complex biological functions of living organisms, many experiments require monitoring of stem cells or bacteria over several generations under different conditions. However, those time-lapse experiments generate large amount of data, which human observers could hardly analyze without bias. Thus, automatated systems for cell tracking are necessary for those studies.

The analysis of the time-lapse microscopic results usually requires not only the tracking of position and locomotion of individual cells, but also the reconstruction of their full lineage. In comparison to pedestrian tracking, the cell tracking task is more challenging due to the constant change of the cellular texture and morphology throughout the cell cycle, the high density of cells with uncertain movement as well as the division events which is not included in other multi-object tracking tasks.

To tackle the cell tracking task, a two-step pipeline consisting of a segmentation/detection phase and an assignment/tracking phase is commonly used. In the first phase, the raw input images are segmented into foreground and background. Those segmentations and the raw data are then fed into classifiers that generate the corresponding detection and division hypotheses. Using the outputs of the classifiers as potentials, a graphical model is built for all possible assignments of detection hypotheses between time frames in the second phase. Such tracking approaches are known as *Tracking-by-assignment* methods, which assume that the previous extracted set of detection hypotheses are over-complete and the constructed model thus describe all tracked targets. A globally consistent tracking solution is then reached via various optimization strategies.

The drawback of such method is that the errors in the first stage would propagate to, and warp, the tracking result. Such errors would occur where a cluster of objects is incorrectly represented by a single segment, termed as *mergers* in this paper. To correct these over- and undersegmentation

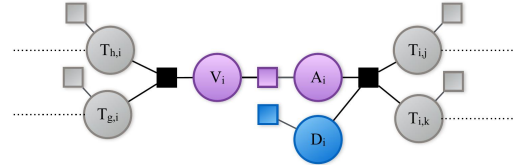


Figure 1. Factor graph for one detection with two incoming and two outgoing transition candidates. The circular nodes represent random variables, the non-black boxes describe factors that are dependent on the connected variables, and filled black boxes represent constraints. The purple detection node i is separated into a disappearance variable V_i and an appearance variable A_i . The binary variable D_i indicates whether object i is about to divide. The transition variables $T \in \{0, \dots, m\}$ indicates the number of objects that are linked between two detection hypotheses.

errors, the Conservation Tracking model which explicitly include the global consistency constraints was developed and was shown to outperform other tracking methods. From previous works, the Conservation Tracking model could be further reformulated into a constrained network flow problem which led to a tight LP relaxation and thus could be solved faster. Even without the constraints for divisions and mergers, the LP relaxation of the model that accounts for flow conservation would still yield integral solutions.

In this paper, we hence developed a iterative method that solve the problem with only the violated constraints identified in each iteration. (a bit more blablabla)

2. Constrained Network Flow Reformulation of Conservation Tracking

2.1. Conservation Tracking Model

The Conservation Tracking model is presented in factor graphs as in Fig. 1. The model contains three types of variables: *Detection* variables $X_i \in \mathcal{X}$ (including the appearance $A_i \in \mathcal{A}$ and disappearance variables $V_i \in \mathcal{V}$) for each connected object from the segmented image, binary *dividing* variables $D_i \in \mathcal{D}$ indicating whether an object is about to divide, and *transition* variables $T_i \in \mathcal{T}$ that represent connections between the detections in two neighboring

time frames. Noteworthy is that a division is allowed if and only if the corresponding detection contains one object. The probabilities of transition are modeled using the center of mass distance between detections, while the probabilities of appearance and disappearance are constant throughout time but linearly decrease to the edge of each image.

Let \mathcal{Y} be the complete set of the configurations of all variables $\mathcal{V} \cup \mathcal{A} \cup \mathcal{T} \cup \mathcal{D}$, the approximate maximum a-posteriori (MAP) solution of the factor graph can be found by minimizing the energy

$$\begin{aligned} y^* &= \arg \max_{y \in \mathcal{Y}} E(y) \\ &= \arg \max_{y \in \mathcal{Y}} \sum_{V \in \mathcal{V}} \sum_{A \in \mathcal{A}} E_x(y_V, y_A) + \sum_{T \in \mathcal{T}} E_T(y_T) + \sum_{D \in \mathcal{D}} E_D(y_D) \end{aligned}$$

subject to constraints for flow conservation, division and mergers which will be explained in ?? in detail. This graphical model can be reformulated into a network flow and solved as an integer linear programming (ILP) problem.

2.2. ILP for Network Flow

To solve ILPs we do LP relaxation.

Our Network flow ILP looks like this:

A LOT OF MATH

2.3. Loosening Constraints

On this we do cutting constraints! Because TUM something.

What we do is cut all constraints and try to solve. If it's solved, great, if not, we add constraints to nodes with flow violation. Then solve again. Repeat until we find valid solution or no new violated nodes.

3. Experiments and Results

Our models are: Drosophila and Rapoport (with and without mergers?)

3.1. Solutions

3.2. Computation Time

4. Conclusion

It works but isn't really worth it.