A Hybrid Approach for Segmentation and Tracking of *Myxococcus Xanthus* Swarms*

Jianxu Chen[1], Shant Mahserejian[2], Mark Alber[2,1], and Danny Z. Chen[1]

[1] Department of Computer Science and Engineering, University of Notre Dame, USA
[2] Department of Appl. and Comput. Math and Stat., University of Notre Dame, USA

Abstract. Segmentation and tracking of moving cells in time-lapse images is an important problem in biomedical image analysis. For *Myxococcus xanthus*, rod-like cells with highly coordinated motion, their segmentation and tracking are challenging because cells may touch tightly and form dense swarms that are difficult to identify accurately. Common methods fall under two frameworks: detection association and model evolution. Each framework has its own advantages and disadvantages. In this paper, we propose a new hybrid framework combining these two frameworks into one and leveraging their complementary advantages. Also, we propose an active contour model based on the Ribbon Snake and Chan-Vese model, which is seamlessly integrated with our hybrid framework. Our approach outperforms the state-of-the-art cell tracking algorithms on identifying completed cell trajectories, and achieves higher segmentation accuracy than some best known cell segmentation algorithms.

1 Introduction

Characterizing the dynamics of cell morphology and cell motility in time-lapse experiments is crucial in various biomedical studies. For example, *Myxococcus xanthus*, a type of Gram-negative bacteria showing distinguished collective motion, was considered a model organism for studying swarming behaviors for decades [2]. Because manual segmentation and tracking are tedious processes with poor reproducibility, there is an increasing need for (semi-)automated methods. In addition, *M. xanthus* poses specific challenges. The obscure boundaries of tightly touching cells, head-to-head touching, and touching in large clusters with high cell density all make cell segmentation difficult (e.g., Fig. 2(a)). The physical distances between cell boundaries may even fall below the imaging resolution. The difficulty in tracking is mainly due to three aspects: the proximity of cell positions; the similarity of nontrivial cell shapes; the diversity of cell behaviors.

Algorithms in the literature for cell segmentation and tracking in time-lapse images can be broadly classified into two categories: detection association (DA) and model evolution (ME). In the DA framework [2,7,8], segmentation is first performed in each image frame, and cell correspondences are built across all

* This work was partly supported by NSF under Grant CCF-1217906 and by NIH under Grants 1R01-GM095959 and 1R01-GM100470.

N. Navab et al. (Eds.): MICCAI 2015, Part III, LNCS 9351, pp. 284–291, 2015.
DOI: 10.1007/978-3-319-24574-4_34

frames of the image sequence into cell trajectories. One advantage of the DA framework is the flexibility in handling various cell behaviors. But, because segmentation is performed separately on each frame, the temporal context is not fully leveraged for segmentation. In some situations, like tightly touching *M. xanthus*, the segmentation may suffer from severe errors for this reason. To some extent, poor segmentation can degrade the tracking accuracy considerably. On the other hand, morphological properties, such as the dynamic changes of cell shapes, cannot be fully extracted from error prone segmentation results.

In the ME framework [3], segmentation and tracking are performed simultaneously by evolving cells in the form of contours from frame to frame, i.e., using the segmentation of one frame to initialize the evolution in the next frame. By propagating information from frame to frame, temporal context is exploited for segmentation and the results could be better than those of the DA framework, especially on images of low quality or when cells have obscure boundaries. But, due to the nature of propagation, such methods may not handle all cell behaviors very well, and may perform poorly when cell displacements are large between consecutive frames. Attempts were made to combine the DA/ME frameworks (e.g., [4]), but such methods can still face troubles when cells are tightly packed.

In this paper, we propose a new hybrid framework for segmenting and tracking cells, which combines the DA and ME frameworks and demonstrates its applicability using *M. xanthus*. In our framework, we first produce a pre-segmentation for each individual frame (the first frame is checked and corrected manually). For each frame K ($K \geq 2$), we develop a *Context Builder* module to work on frames $K-1$ to $K+t$ ($t > 1$ is constant) to build cell correspondence (similar to the DA framework), and confirm cells newly entering frame K and assign trajectory identities to cells with *reliable pre-segmentation* in frame K. Next, we apply a *Cell Tracker* module to perform an active contour evolution to segment and track the remaining cells in frame K (similar to the ME framework). The spatial-temporal context extracted from *Context Builder* is utilized to guide this contour evolution. To handle *M. xanthus* specifically, we propose an open active contour model based on Ribbon Snake [6] and the Chan-Vese model [1].

Our hybrid framework takes advantage of the complimentary strengths of the DA and ME frameworks. Our *Context Builder* extracts and analyzes the temporal context to provide information for *Cell Tracker* to guide cell evolution. The *Cell Tracker* tackles the segmentation and tracking of those cells where pre-segmentation and association fail; it refines the pre-segmentation so that *Context Builder* can build more accurate correspondence for the next frame. Consequently, our method obtains better tracking results than the state-of-the-art cell tracking algorithms, and achieves higher segmentation accuracy than some best known cell segmentation algorithms. Table 1 summarizes our results.

2 Methodology

The main steps of our framework are shown in Fig. 1. On the input image sequence, pre-segmentation is produced for each frame. The segmentation algorithm used can be application-dependent; the one we use for *M. xanthus* is

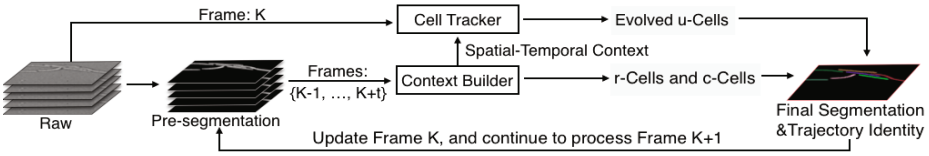

Fig. 1. An outline of our proposed framework: Pre-segmentation is first obtained. For each frame K, *Context Builder* extracts the spatial-temporal context. In frame K, reliable cells (r-Cells) are associated with existing trajectories, and entering/re-appearing cells are confirmed (c-Cells). Cells in frame $K-1$ with uncertain correspondence (u-Cells), together with the spatial-temporal context, are fed to *Cell Tracker* to find their positions in frame K. The c-Cells, r-Cells, and evolved u-Cells are merged to produce the final segmentation and trajectory identity of each cell in frame K.

described in Section 3. The first frame is manually examined and corrected. Note that it is easy to make our method fully automated. Namely, the first appearance of each cell can be confirmed by temporal context (called c-Cells, as discussed in Section 2.1). Manual correction is especially useful in low quality images, so that additional prior knowledge can be exploited to tackle extremely difficult cases.

For each frame $K \geq 2$, our method works in two steps. In step 1, the *Context Builder* employs a matching model based on the Earth Mover's Distance (EMD) [2] on frames $\{K-1, \ldots, K+t\}$, where $t > 1$ is a constant indicating the depth of the temporal context (we use $t = 3$). The objectives of *Context Builder* are: (1) detect and confirm the newly entering cells to frame K; (2) confirm cells in frame K that have good pre-segmentation and associate them with some existing trajectories; (3) prepare the spatial-temporal context information for *uncertain cells* (called u-Cells). Intuitively, u-Cells are cells in frame $K-1$ whose corresponding cells in frame K are hard to segment or associate with. In step 2, the *Cell Tracker* segments and tracks u-Cells by active contours controlled by three external forces: the image force, shape-prior force, and repelling force. For *M. xanthus* specifically, we propose an open active contour model based on Ribbon Snake [6] and the Chan-Vese model [1]. At the end, the final segmentation of frame K is obtained; each segmented cell in frame K either is associated with an existing trajectory or starts a new trajectory. Note that *M. xanthus* cannot merge. Yet, our framework is easy to adjust for cell fusion in other problems, by allowing some contours to merge in Cell Tracker.

2.1 Context Builder

Our *Context Builder* module works on consecutive frames $K-1$ to $K+t$ to build the temporal context by performing a hierarchical matching based on the Earth Mover's Distance (EMD) matching model [2]. The major advantages of the EMD matching model are the capability of handling cells moving in/out frames and cell divisions, and the robustness to various segmentation errors [2]. Suppose frame $K \geq 2$ is being processed and trajectories have been constructed in frames 1 to $K-1$. The matching model is applied to establish cell correspondence in

frames $K-1$ to $K+t$. Frame $K-1$ is included in order to associate cells in frame K with some existing trajectories. By analyzing the cell correspondence in the temporal context, three types of objects will be extracted: confirmed cells (c-Cell), reliable cells (r-Cells), and uncertain cells (u-Cell).

C-Cells are cells in frame K that have no correspondence in frame $K-1$ but have corresponding cells with consistent shapes in frames $K+1,\ldots,K+t$. Confirmed by the temporal context, c-Cells could be entering cells with high likelihood of not being false alarm. So, manual correction is not needed for entering cells. A cell P in frame K is an r-Cell if P has a one-to-one correspondence in frame $K-1$ with consistent body shape. The pre-segmentation of P can be finalized, due to the reliable correspondence and the high shape similarity. U-Cells are cells in frame $K-1$ with no correspondence in frame K or whose corresponding cells in frame K are not r-Cells. Intuitively, u-Cells are cells at the ends of some existing trajectories, whose correspondence in frame K cannot be confidently determined. If a u-Cell is touching the image border and has no correspondence in succeeding frames, its trajectory will be terminated (i.e., considered as leaving cells). The remaining u-Cells are propagated to frame K and fed to *Cell Tracker*. The final segmentation of frame K contains all successfully evolved u-Cells, all r-Cells, and c-Cells not overlapping with any evolved contours.

The *Context Builder* also prepares the spatial-temporal context for the *Cell Tracker*. It forms a *Barrier Map* for all r-Cells, which will be used to define the repelling force in *Cell Tracker* (see Section 2.2). Moreover, it uses a *Contour Initializer* that acts on all propagated u-Cells to initialize their evolution for *Cell Tracker*. For a u-Cell Q, an initial contour in frame K is obtained by Earth Mover's Morphing (EMM) [5]. If Q has at least one corresponding cell in frame K, then EMM is applied to Q and all its corresponding cells in frame K. If Q has no corresponding cell in frame K, but at least one corresponding cell in frames $K+1$ to $K+t$, then EMM is applied to Q and the corresponding cell with the highest reliability (computed by the EMD matching model). Then, the position of Q in frame K can be extracted from the morphing result. Note that multiple u-Cells sharing common corresponding cells will be analyzed together to ensure consistent results. If no cell is associated with Q and Q is away from the image border, then Q is directly copied to frame K as the initial contour.

2.2 Cell Tracker

The *Cell Tracker* module employs an open active contour model to segment and track cells simultaneously. The initial positions of the contours are obtained from u-Cells by *Contour Initializer*. The evolution of each contour ϕ is governed by the following equation (we use $\omega_1 = \omega_2 = \omega_4 = 1$ and $\omega_3 = 0.25$):

$$\frac{\partial \phi}{\partial t} = \mathbf{F}\|\nabla\phi\|, \text{ where } \mathbf{F} = \omega_1\mathbf{F}_{image} + \omega_2\mathbf{F}_{internal} + \omega_3\mathbf{F}_{repel} + \omega_4\mathbf{F}_{shape} \quad (1)$$

For *M. xanthus*, the contours are represented by Ribbon Snake [6]. Each cell contour has the parametric form $\phi(s) = (x(s), y(s), w)$, where $(x(s), y(s))$ is a

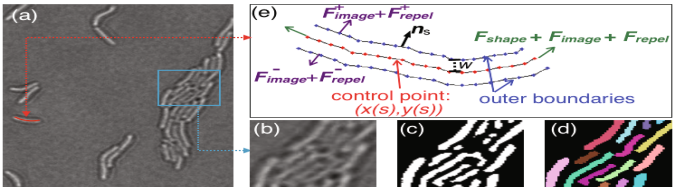

Fig. 2. (a) An input image of *M. xanthus* cells; (b) zoom-in of a clustered region; (c) segmentation of [9]; (d) segmentation of our method; (e) illustrating Ribbon Snake.

parametric curve, called the *centerline*, and w is the half width of the contour, which is computed from its segmentation result in the preceding frame. The concept is illustrated in Fig. 2(e). For each control point $(x(s), y(s))$, the image force (resp., repelling force) is the sum of the image forces (resp., repelling forces) exerted on the two outer boundaries of the contour in the normal direction. Namely, $\mathbf{F}_{image} = \mathbf{F}^+_{image} + \mathbf{F}^-_{image}$, and $\mathbf{F}_{repel} = \mathbf{F}^+_{repel} + \mathbf{F}^-_{repel}$. Here, $\mathbf{F}^+_*$ (resp., $\mathbf{F}^-_*$) is the force applied to $(x(s), y(s)) + w\mathbf{n}_s$ (resp., $(x(s), y(s)) - w\mathbf{n}_s$), where $\mathbf{n}_s$ is the unit normal vector at $(x(s), y(s))$. Additionally, for the two endpoints of the centerline, extra forces are applied in the tangential direction, including $\mathbf{F}_{shape}$, $\mathbf{F}_{image}$, and $\mathbf{F}_{repel}$. For clarification, at the two tips, the tangential component of $\mathbf{F}_{image}$ and $\mathbf{F}_{repel}$ is computed at the endpoints of the centerline, while the normal component is calculated based on the forces at the corresponding points on the outer boundaries. Finally, $\mathbf{F}_{internal}$ acts on all control points to regulate the contour smoothness, with the classic form $\mathbf{F}_{internal} = \alpha\phi'' - \beta\phi''''$, where α and β are weighting parameters (we use $\alpha = 0.4$, $\beta = 0.2$). Below, we explain the calculation of $\mathbf{F}^+_{image}$, $\mathbf{F}^+_{repel}$, and $\mathbf{F}_{shape}$. The others can be obtained similarly.

The image force is computed using the Chan-Vese model [1]. Specifically, $\mathbf{F}^+_{image} = -((I^+ - \mu_1)^2 - (I^+ - \mu_2)^2)\mathbf{n}_s$, where I^+ is the intensity at $(x(s), y(s)) + w\mathbf{n}_s$, μ_1 is the average intensity inside the cell region enclosed by the outer boundaries, μ_2 is the average intensity outside all cell regions.

A contour will suffer from the repelling force defined by the *Barrier Map* or when it moves close to other contours. Basically, the *Barrier Map* determines the magnitude of the repelling force from r-Cells to any point in the image region. Suppose $\{\phi_1, \phi_2, \ldots, \phi_n\}$ are all evolving contours. For a point on the outer boundary of ϕ_i, say $(x(s), y(s)) + w\mathbf{n}_s$, $\mathbf{F}^+_{repel}(s) = -[F^+_{bm}(s) + \sum_{j \neq i} F^+_{ij}(s)]\mathbf{n}_s$. The force from *Barrier Map* is $F^+_{bm}(s) = 1/[1 + exp(2d^+(s) - d_{max})]$, if $0 < d^+(s) \leq d_{max}$; $F^+_{bm}(s) = \infty$, if $d^+(s) = 0$; otherwise, $F^+_{bm}(s) = 0$. Here, $d^+(s)$ is the shortest distance from $(x(s), y(s)) + w\mathbf{n}_s$ to any points on the skeletons of r-Cells, and d_{max} is a constant cut-off value of the furthest distance that the repelling force will act on. As a rule of thumb, d_{max} can be set as the maximum cell thickness. Moreover, the force from contour ϕ_j $(j \neq i)$ is $F^+_{ij}(s) = 1/[1 + exp(2d^+_j(s) - d_{max})]$, if $0 < d^+_j(s) \leq d_{max}$; $F^+_{ij}(s) = \infty$, if $d^+_j(s) = 0$; otherwise, $F^+_{ij}(s) = 0$. Here, $d^+_j(s)$ is the distance from $(x(s), y(s)) + w\mathbf{n}_s$ to ϕ_j, and d_{max} is the same as above.

Between consecutive frames, the shape of each cell should be consistent. In terms of *M. xanthus*, certain lengths can be expected (except for cell divisions).

Inspired by [3], we take such information as the shape prior and impose into the evolution. Specifically, the initial contours are shortened by 40% in order to reduce the possibility of part of an initial contour residing outside the cell body. Then, $\mathbf{F}_{shape}$ is imposed on the two tips of the contour in the tangential direction outwards to make the cell growing into the expected length. Suppose the length of a particular cell P in frame $K-1$ is L^{K-1}. Then the target length of P in frame K, denoted by L_T^K, is set as L^{K-1}. The magnitude of $\mathbf{F}_{shape}$ is defined as $V_{shape}^{(i)} = \gamma^{(i)} * ds$, where $i \in \{1, 2\}$ is the index of the two tips. Suppose the current length of the contour is L. $ds = 1$ if $L < 0.85L_T^K$; $ds = 0.5$ if $0.85L_T^K \leq L < 0.95L_T^K$; $ds = 0$ if $0.95L_T^K \leq L < 1.2L_T^K$; $ds = -1$ if $L \geq 1.2L_T^K$. The value of γ depends on the image force and repelling force. Suppose v_1 and v_2 are the magnitudes of the tangential component of $\mathbf{F}_{image} + \mathbf{F}_{repel}$ at the two tips, respectively. If $v_1 \geq 0$ and $v_2 \geq 0$, then $\gamma^{(1)} = \gamma^{(2)} = 1$. If $v_1 < 0$ and $v_2 \geq 0$, then $\gamma^{(1)} = 0$, $\gamma^{(2)} = 1$ (similar for $v_1 \geq 0$, $v_2 < 0$). If v_1 and v_2 are both negative, suppose $v_1 > v_2$, then $\gamma^{(1)} = 2$, $\gamma^{(2)} = 0$. In general, $v_i > 0$ indicates expansion, and $v_i < 0$ means the contour will be compressed on that tip. Thus, when not both v_i's are positive, the shape force will only be exerted on the tip that will not be compressed; when v_i's are both negative, the shape force will only act on the tip suffering less compression. Finally, $\mathbf{F}_{shape}^{(i)} = V_{shape}^{i}\mathbf{t}_i$, where $\mathbf{t}_i$ is the outward tangential vector at the centerline endpoints.

There are two special cases for $\mathbf{F}_{shape}$ exerted on P. (I) If P touches the image border, $\mathbf{F}_{shape}$ is voided. In this situation, the length restriction may make contours leak to the background (when part of a cell moves out of the image region), or may make a contour not grow to the actual length (when the cell is gradually moving into the image region). (II) If the length of P after the evolution in frame K is shorter than $0.9 * L_T$, and P needs to evolve again in frame $K+1$, then the target length in frame $K+1$, i.e., L_T^{K+1}, will be set as L_T^K, not L^K. In this situation, if we always take the actual length in the preceding frame as the target length, a cell may keep shrinking frame by frame.

Finally, *Cell Tracker* checks for potential cell divisions, since *M. xanthus* may split in the middle of a cell body. After the evolution of each contour, the intensity of the centerline in the grayscale image is extracted. A local regression with the second degree polynomial is performed to smooth the values. If the minimum is located near the middle of the centerline, and the ratio of the maximum to the minimum is larger than a threshold defined experimentally, the contour will be split in the middle. The contours of the daughter cells will evolve again to find the optimal positions. Meanwhile, the cell lineage can be established. There are many advanced division detection methods in the literature, such as the training-based approach in [8]. Our trial studies showed that our division checking is effective for our problem, thus adopted for the sake of simplicity.

3 Experiments and Evaluations

We conducted experiments on seven image sequences of *M. xanthus* obtained in our lab. In general, they were all acquired by a digital camera attached to a

Table 1. Tracking and segmentation results on medium (M), low (L), high (H1-H4) quality data, and special (S) data with large cell displacements in consecutive frames.

Datasets			Tracking Results					Segmentation Accuracy			
	Traj. #	Frame #	Our	[3]	[2]	[7]	[8]	Our	Otsu	[9]	[10]
M	79	71	**70**	18	64	46	61	**0.96**	0.82	0.69	0.61
L	50	69	**41**	21	31	19	19	**0.91**	0.61	0.78	0.64
S	74	47	**67**	5	59	32	21	**0.95**	0.75	0.82	0.78
H1	168	356	**148**	57	119	79	88	**0.95**	0.77	0.92	0.87
H2	104	60	**102**	80	99	78	80	**0.95**	0.73	0.93	0.88
H3	113	100	**110**	76	98	88	92	**0.97**	0.73	0.91	0.89
H4	112	90	**106**	75	96	87	80	**0.94**	0.76	0.89	0.91

microscope with 60x oil-immersion objectives. The high quality dataset (H1-H4) were acquired with a faster camera. The metric for the tracking performance is the number of *completed trajectories*. This is a harsh metric because merely a few error correspondences in tracking may result in huge loss in completed trajectories. All results were examined visually by human experts. Our method is compared against four state-of-the-art algorithms [3,2,8,7]. [3] is a representative method under the model evolution framework. The other three are different detection association approaches. [2] is a hierarchical scheme based on the EMD matching model. [8] conducts global optimization across the whole image sequence. [7] performs frame-wise optimal matching. The pre-segmentation of our method was obtained in two steps: (i) A pixel-wise classification is performed using [9]; (ii) the detected regions were cut at the intersection points of the region centerlines or at each centerline point whose curvature is larger than an experimentally determined threshold. For fair comparison, the same pre-segmentation was used for our method and for [2] and [7]. The segmentation of [8] was also obtained by [9], but an object count classifier [8] was applied instead of explicitly cutting the regions. Note that no human intervention was made in our method except manually correcting the segmentation in the first frame. However, the model evolution method [3] may need other human interventions as discussed in [3]. All results are shown in Table 1. In dataset S, the same cell in two consecutive frames may overlap less than 50%, which showed that our method is more effective than the model evolution method [3] in dealing with large cell displacements. Also, the results showed that our method has better performance than the three detection association approaches, even in low quality images (L).

Further, we compared the segmentation results with three segmentation methods to show the advantages of our hybrid framework on segmentation, especially the benefits of leveraging temporal context in segmentation. First, the Otsu thresholding is used as the baseline method. Second, ilastik [9] is one of the best open source software for cell segmentation based on pixel classification. Finally, the method in [10] is one of the most effective cell segmentation algorithms based on correlation clustering. Except ours, the other methods are all performed on individual image frames. We manually segmented 20 frames randomly selected

in each dataset as the ground truth. Suppose there are N true cells in the manual segmentation, denoted by $\{G_1, G_2, \ldots, G_N\}$, and there are M segmented cells, denoted by $\{O_1, O_2, \ldots, O_M\}$. A segmented cell O_k is a true positive (TP), if $\exists i$ s.t. $\|G_i \cap O_k\| > 0.5 * \|G_i\|$ and $\|G_i \cap O_k\| > 0.5\|O_k\|$. Then, the precision is $Pr = TP/M$, and the recall is $Re = TP/N$. Finally, the segmentation accuracy is measured by F1-score, i.e., $F = 2 * Pr * Re/(Pr + Re)$. An example of segmentation in the low quality dataset (L) is shown in Fig. 2(b)-(d). The results in Table 1 showed our hybrid framework achieves much higher segmentation accuracy than the other methods without utilizing temporal context.

4 Conclusions

In this paper, we propose a new hybrid framework for segmentation and tracking of moving cells in time-lapse images. In addition, a new active contour model is proposed for *M. xanthus*, and is seamlessly integrated with our hybrid framework. The evaluation shows that the proposed method outperforms state-of-the-art cell tracking algorithms and some best known cell segmentation algorithms.

References

1. Chan, T.F., Vese, L.A.: Active contours without edges. IEEE Trans. Image Processing 10(2), 266–277 (2001)
2. Chen, J., Harvey, C.W., Alber, M.S., Chen, D.Z.: A matching model based on earth mover's distance for tracking *myxococcus xanthus*. In: Golland, P., Hata, N., Barillot, C., Hornegger, J., Howe, R. (eds.) MICCAI 2014, Part II. LNCS, vol. 8674, pp. 113–120. Springer, Heidelberg (2014)
3. Deng, Y., Coen, P., Sun, M., Shaevitz, J.W.: Efficient multiple object tracking using mutually repulsive active membranes. PloS One 8(6), e65769 (2013)
4. Li, K., Miller, E.D., Chen, M., Kanade, T., Weiss, L.E., Campbell, P.G.: Cell population tracking and lineage construction with spatiotemporal context. Medical Image Analysis 12(5), 546–566 (2008)
5. Makihara, Y., Yagi, Y.: Earth mover's morphing: Topology-free shape morphing using cluster-based EMD flows. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010, Part IV. LNCS, vol. 6495, pp. 202–215. Springer, Heidelberg (2011)
6. Mayer, H., Laptev, I., Baumgartner, A.: Multi-scale and snakes for automatic road extraction. In: Burkhardt, H.-J., Neumann, B. (eds.) ECCV 1998. LNCS, vol. 1406, pp. 720–733. Springer, Heidelberg (1998)
7. Padfield, D., Rittscher, J., Roysam, B.: Coupled minimum-cost flow cell tracking for high-throughput quantitative analysis. Med. Image Anal. 15(4), 650–668 (2011)
8. Schiegg, M., Hanslovsky, P., Kausler, B.X., Hufnagel, L., Hamprecht, F.A.: Conservation tracking. In: ICCV, pp. 2928–2935 (2013)
9. Sommer, C., Straehle, C., Kothe, U., Hamprecht, F.A.: ilastik: Interactive learning and segmentation toolkit. In: ISBI, pp. 230–233 (2011)
10. Zhang, C., Yarkony, J., Hamprecht, F.A.: Cell detection and segmentation using correlation clustering. In: Golland, P., Hata, N., Barillot, C., Hornegger, J., Howe, R. (eds.) MICCAI 2014, Part I. LNCS, vol. 8673, pp. 9–16. Springer, Heidelberg (2014)