Published in final edited form as:

IEEE Trans Med Imaging. 2016 September ; 35(9): 2074–2084. doi:10.1109/TMI.2016.2548490.

A Hybrid Approach for Segmentation and Tracking of Myxococcus Xanthus Swarms

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

[1]Department of Computer Science and Engineering, University of Notre Dame, US

[2]Department of Applied and Computational Mathematics and Statistics, University of Notre Dame, US

Abstract

Cell segmentation and motion tracking in time-lapse images are fundamental problems in computer vision, and are also crucial for various biomedical studies. *Myxococcus xanthus* is a type of rod-like cells with highly coordinated motion. The segmentation and tracking of *M. xanthus* are challenging, because cells may touch tightly and form dense swarms that are difficult to identify individually in an accurate manner. The known cell tracking approaches mainly fall into two frameworks, detection association and model evolution, each having 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, which is seamlessly integrated with our hybrid framework. Evaluated by 10 different datasets, our approach achieves considerable improvement over the state-of-the-art cell tracking algorithms on identifying complete cell trajectories, and higher segmentation accuracy than performing segmentation in individual 2D images.

I. Introduction

Characterizing the dynamics of cell morphology and motility is a fundamental problem in various biomedical studies. With the development of advanced techniques on imaging and experiments, identifying the trace of cell populations in both the space and time in time-lapse images has become an important issue in computer vision [1]. For example, cell tracking algorithms have been applied to quantitative biological studies [2], such as quantifying the effects of inhibitors on cells passing through the cell-cycle, monitoring cell motility, and measuring the nuclear translocation. Also, tracking cells and the evolution of the associated constellation patterns during the development of kidney tissues is a key step in the study of cystic kidney diseases [3], [4].

The objective of this work is the segmentation and tracking of *Myxococcus xanthus*, a social surface-dwelling bacterium. The rod-like *M. xanthus* cells are a type of Gram-negative bacteria with noticeable collective motions, called swarming. Cell-cell contact occurs frequently to regulate individual behaviors while the whole swarm is constantly in motion [5]. Time-lapse confocal microscopy is widely used to visualize the development of *M. xanthus* swarms over time. Experiments have been conducted to investigate the mechanism of *M. xanthus* swarming at different scales, from the individual cell's surface gliding

motility [6] to the highly coordinated motions in large swarms [7]. *M. xanthus* has been recognized as a model organism for studying the collective cell motion principles, which may be generalized to other cellular systems [8].

The time-lapse images collected in an experiment may contain hundreds of image frames with hundreds of cells in each frame. Thus, manual cell segmentation and tracking tends to be a tedious process with poor reproducibility, which dramatically increases the need for (semi-) automated methods.

M. xanthus poses new challenges for segmentation and tracking. During cell-cell collision, the physical distances between cell boundaries may fall below the imaging resolution. The obscure boundaries of tightly touching cells — head-to-head touching and touching in large clusters with high cell density — make cell segmentation difficult (e.g., Fig. 1). Furthermore, over-segmentation or false negative errors are sometimes difficult to avoid within tightly packed swarms. This is because the squeezed space among individual cells may cause local image degradation, and consequently make it difficult to correctly detect all cells. The difficulty in tracking is mainly due to four aspects: the proximity of cell positions; the similarity of elongated cell shapes; the diversity of cell behaviors; the cells being difficult to detect.

A. Related Work

Known algorithms for cell segmentation and tracking on time-lapse images can be broadly classified into two categories: detection association (DA) and model evolution (ME).

1) Detection Association—In the DA framework, segmentation is first performed on each image frame, and optimal cell correspondences are then built across all frames of the image sequence to form cell trajectories. The optimality of cell correspondences can be defined locally, frame-wise, or globally. Local optimization is used to search for the best match for a particular cell within a certain spatial range in the next frame [9]. Optimal associations between cells in consecutive frames can be formulated as a minimum-cost flow problem [10] or 0–1 integer programming [11]. Also, the correspondences in the whole image sequence can be obtained by a global optimization using the Viterbi algorithm [12] or *k*-partite graph matching [13]. A hierarchical approach was also proposed, which performs both frame-wise and global optimization [14].

Joint segmentation and tracking schemes were proposed recently [4], [15]. Different from traditional detection association methods, the detection step only obtains segmentation candidates. In the association step, the actual segmentation and trajectories are built simultaneously in a single model, which optimizes spatial-temporal consistency globally.

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 when obtaining the segmentation. In some situations, like tightly touching *M. xanthus*, the segmentation may suffer from severe errors for this reason. Poor segmentation can degrade the tracking accuracy considerably, although some approaches may handle certain types of segmentation errors [14], [15], [16]. On the other hand,

morphological properties, such as dynamic changes of cell shapes, cannot be fully extracted from error-prone segmentation results.

2) Model Evolution—In the ME framework, 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. Mean-shift based algorithms have been used to track migrating cells when only cell centroids are of interest [17]. Active meshes [18], open active contours [19], and level set based techniques [20], [21] have been applied to track various types of cells. Shape constraints [22] or topology constraints [23] can be adopted during evolution to achieve higher accuracy, and various techniques [21], [24] can be used to reduce the computational complexity of conventional level sets significantly. Also, model evolution can be extended and implemented in the graph-cut framework [25], [26].

But, due to the nature of propagation, ME methods cannot handle all cell behaviors well enough, and may perform poorly when cell displacements are large between consecutive frames. For example, a separate step is usually applied to each frame, before or after evolution, to identify cells entering the imaging window using classic segmentation methods [20], [24], [25]. Understandably, this segmentation step could be error-prone, which may introduce false contours and propagate errors. Moreover, ME methods often assume that the same cell overlaps considerably in consecutive frames.

Attempts have been made to combine the DA and ME frameworks (e.g., [27]); but, such methods still face troubles when cells are tightly packed.

B. Our Contributions

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 on time-lapse images of *M. xanthus*. The spirit of our approach is to employ a detection association scheme to build preliminary correspondence, then analyze the temporal correspondence to identify cells whose segmentation and tracking may be problematic, and finally invoke a model evolution scheme to handle those troublesome cells.

Our hybrid framework takes advantage of the complementary strengths of the DA and ME frameworks by means of two complementary modules, called *Context Builder* and *Cell Tracker*.

The *Context Builder* module (similar to the DA framework) extracts and analyzes the spatial-temporal context to guide the contour evolution performed by the *Cell Tracker*. The guidance from the context helps overcome common issues in known active contour methods, e.g., leakage or inability to deal with large cell displacements between consecutive frames. Also, the *Context Builder* can alleviate the burden of the contour evolution in handling entering cells, which is another common limitation of the ME framework. In addition, when using only the ME framework, each contour has to evolve in every frame. In contrast,

combining with the DA framework allows a contour to evolve only when necessary. This may reduce the possibility of error propagation, comparing to the ordinary ME framework.

The *Cell Tracker* module (similar to the ME framework) tackles the segmentation and tracking of those cells on which pre-segmentation or association fails; it refines the pre-segmentation so that the *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 well-known cell segmentation algorithms which work on each individual frame.

This work extends our pioneer work in [28]. The core idea of this paper, i.e., a systematical scheme combining the DA and ME frameworks, was sketched in [28]. Also, some preliminary experiments showed promising results. In this paper, we propose a new matching scheme for the *Context Builder* in the framework, while the work in [28] directly adopted the EMD matching model in [14]. We formulate the matching model in a different way and design a new measure as the EMD ground distance, a key component of the model. As a consequence, our new algorithm can establish a more accurate temporal context than that in [28] (see Section II-A1 for detailed discussion). Besides, it was suggested in [28] to manually correct the segmentation of the first frame. Here, we propose a new approach to automatically refine the segmented regions in the first frame, which achieves comparable performance as manual correction. Moreover, a thorough evaluation of our proposed approach is conducted with more diverse datasets. We systematically evaluate the stability of parameter selections, segmentation and tracking accuracy. A novel measurement is proposed in this work for segmentation accuracy evaluation. Finally, we discuss both the limitations and applicability of our proposed framework in various cell tracking problems.

C. Organization of the Paper

The rest of the paper is organized as follows. Section II describes the proposed hybrid framework, which includes two key modules, *Context Builder* (Section II-A) and *Cell Tracker* (Section II-B). Section III presents the implementation details, evaluation and comparison results. Finally, Section IV concludes the paper and discusses the generality of the proposed hybrid framework and future work.

II. Methodology

Fig. 2 shows our framework outline. On each frame of the input image sequence, pre-segmentation is obtained by an application-dependent algorithm. The one we use for *M. xanthus* is described in Section III.

The pre-segmentation of the first frame may be manually examined and corrected. Manual correction is especially useful for low quality images, serving as prior knowledge to tackle extremely difficult cases. Detailed discussion on manual correction and automation will be presented in Section III-C.

On each frame $K > 1$, our method performs two steps:

Step 1: The *Context Builder* employs a matching model based on the Earth Mover's Distance (EMD) on frames $\{K-1; \ldots; K+t\}$ to associate detected cells, where the constant $t > 1$ indicates the depth of the temporal context. The tasks of the *Context Builder* are: (1) detecting and confirming the newly entering cells in frame K; (2) confirming cells in frame K that have good pre-segmentation and associating them with some existing trajectories; (3) identifying the cells on which segmentation or association is uncertain. For such error-prone cells, the *Cell Tracker* will be invoked in the next step.

Step 2: The *Cell Tracker* simultaneously segments and tracks the error-prone cells identified in Step 1, using active contours. For *M. xanthus* specifically, we propose an open active contour model based on the Ribbon Snakes [29] and the Chan-Vese model [30]. Note that while *M. xanthus* cannot merge, the Cell Tracker can be adapted to cell fusion in other problems, by allowing contours to merge during the evolution.

At the end, the final segmentation of frame K is obtained. Each segmented cell in frame K is either associated with an existing trajectory or starts a new trajectory.

A. Context Builder

Our *Context Builder* module works on consecutive frames $K-1$ to $K+t$ to build the temporal context by performing hierarchical matching based on the Earth Mover's Distance (EMD) matching model [14]. The major advantages of the EMD matching model include the capability of handling cells moving in/out the image frames and cell divisions, and the robustness to various segmentation errors, such as false negative segmentations or falsely merged cells.

Suppose frame $K \geq 2$ is being processed and cell trajectories have been constructed for frames 1 to $K-1$. The EMD matching model is applied to establish cell correspondences in frames $K-1$ to $K+t$ (we use $t = 3$, so the depth of the temporal context is five consecutive frames). Frame $K-1$ is included in order to associate cells in frame K with some existing cell trajectories. By analyzing the cell correspondences 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 the cells in frame K that have no correspondence with the cells in frame $K-1$ but have corresponding cells with consistent shapes (i.e., the length differs less than a parameter L_{enter}) and positions 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. Thus, no manual correction is needed for entering cells in our method.

R-Cells are the cells in frame K that have unique one-to-one correspondence with some cells in frame $K-1$ with consistent cell shapes (i.e., the length differs less than L_{seg}). The pre-segmentation of r-Cells can be finalized, due to the reliable correspondence and the high shape similarity with cells in frame $K-1$.

U-Cells are the 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 the cells at the ends of some existing trajectories, whose correspondence cannot be confidently determined in frame K. If a u-Cell touches the image border and has no correspondence in the succeeding frames, then its trajectory will be terminated (i.e., considered as a leaving cell). 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.

1) Matching Algorithm—We adopt and re-formulate the hierarchical association scheme based on the EMD matching model in [14] for linking the detected cells. The classical definition of EMD can be found in [14] and [31].

EMD Matching Model and Hierarchical Association: We extend the classical definition of EMD to a matching model, and our hierarchical association scheme applies this matching model at two levels, namely, between every two consecutive frames and within a certain number of consecutive frames (i.e., the temporal context).

In our EMD matching model, the goal is to find the optimal correspondence between two sets of cells, $\mathcal{P} = \{(p_1, w_{p_1}), \ldots, (p_n, w_{p_n})\}$ and $\mathcal{Q} = \{(q_1, w_{q_1}), \ldots, (q_m, w_{q_m})\}$, where p_i (resp., q_j) is a cell with w_{p_i} (resp., w_{q_j}) pixels in an image. Two virtual cells p_0 and q_0 with very large weights, e.g., $w_{p_0} = w_{q_0} = \infty$, are also created.

The model is formulated as the following optimization problem.

$$EMD(\hat{\mathcal{P}}, \hat{\mathcal{Q}}) = \min_{f_{ij}} \sum_{0 \le i \le n} \sum_{0 \le j \le m} D_{ij} f_{ij} \quad (1)$$

subject to

$$\sum_{0 \le j \le m} f_{ij} = w_{p_i}, \forall 1 \le i \le n \quad (2a)$$

$$\sum_{0 \le i \le n} f_{ij} = w_{q_j}, \forall 1 \le j \le m \quad (2b)$$

$$f_{ij} \ge 0, \forall 0 \le i \le n, 0 \le j \le m \quad (2c)$$

Here, $\hat{\mathcal{P}} = (p_0, w_{p0}) \cup \mathcal{P}$ and $\hat{\mathcal{Q}} = (q_0, w_{q0}) \cup \mathcal{Q}$. Optimal cell correspondences between $\mathcal{P}$ and $\mathcal{Q}$ can be obtained by solving the above optimization problem, in the sense that the minimum EMD value means the smallest total matching cost and a large f_{ij} value (for $i \neq 0$ and $j \neq = 0$) implies a strong correspondence between p_i and q_j.

An intuitive interpretation of the EMD model is that p_i is a pile of dirt of w_{p_i} quantities, and q_j is a hole which can hold dirt by up to w_{q_j} quantities; f_{ij} is the amount of dirt distributed from pile p_i to hole q_j, and D_{ij} is the cost for dirt distribution of one unit quantity from p_i to q_j. Then, EMD is to compute the minimum total cost for the entire dirt distribution.

D_{ij}, also called the *ground distance*, is a key component of the EMD model. Our definition and computation of D_{ij} are detailed in Appendix 2. Intuitively, D_{ij} has the following interpretation. When i and j are both positive, D_{ij} measures the matching cost of two cell detections. $D_{00} = \infty$, because flows between virtual cells are meaningless. If either i or j is 0, but not both, there are two different interpretations. Suppose $i > 0$ and $j = 0$. If p_i is close to the image border, then D_{ij} measures the cost for p_i leaving the imaging window. Otherwise, D_{ij} is the cost for a pixel in p_i matching to nothing, which is used to model the situation that p_i is a false alarm or the corresponding cell is somehow missing. Also, this is designed to accommodate the possible discrepancy between the numbers of pixels in the segmentation of the same cell across different frames. The case with $i = 0$ and $j > 0$ can be interpreted in a similar manner.

We apply the EMD matching model in a hierarchical fashion. In the low-level association (Fig. 3(b)), the EMD matching model is applied to every two consecutive frames in a conservative manner. In other words, we accept only strong correspondence, i.e., $p_i \in \mathcal{P}$ is associated with $q_j \in \mathcal{Q}$ only if the f_{ij} value in the optimal solution satisfies $f_{ij} > \rho \cdot \min(w_{p_i}, w_{q_j})$ and $D_{ij} < \lambda$. Here, ρ and λ are determined empirically.

Child-deficient (resp., *parent-deficient*) cells: For a cell in frame K, its corresponding cells in frame $K+1$ (resp., $K-1$) are called *children* (resp., *parents*). For each cell not in the last (resp., first) frame of the temporal context, if the total weight sum of its children (resp., parents), W_{sum}, and its own weight, W_{cell}, satisfy $|W_{sum} - W_{cell}| > (1-\rho)|W_{cell}|$, then we call it *child-deficient* (resp., *parent-deficient*). Here, ρ is the same as the aforementioned parameter, i.e., the maximum acceptable ratio between the flow amount and cell size.

In the high-level association (Fig. 3(c)), the EMD matching model is applied to all *child-deficient* cells and *parent-deficient* cells within the temporal context, i.e., frames $K-1$, K, …, $K+t$. The ground distance is assigned using the same method as for the low-level association. Different from the low-level, the weight of each cell is set as $|W_{sum} - W_{cell}|$. In the optimal solution, the correspondence between a child-deficient cell and a parent-deficient cell actually reconnects a “broken” trajectory.

Note that, the EMD matching model in [14] only requires the total amount of flows not exceeding the total weight of P and that of Q, but, when the total weights are highly unbalanced between P and Q, only a small amount of flows is allowed, and thus some of the actual correspondence cannot be established. Comparing to [14], the EMD matching model is improved here. Specifically, we require that all “dirt” of each pile be completely

distributed to either some actual holes or the virtual hole (q_0), and the "space" of each hole be fully filled by flows from either some actual piles of dirt or the virtual pile (p_0).

2) Spatial-temporal Context—The *Context Builder* assists the *Cell Tracker* by preparing the spatial-temporal context. The spatial context forms a *Barrier Map* of all r-Cells, which will be used to partially guide the contour evolution in the *Cell Tracker* (see Section II-B2), namely, defining a force field according to the positions of all r-Cells. The force field is of large magnitude when close to r-Cells, and vanishes when far enough away from the vicinity of r-Cells.

The temporal context acts on all propagated u-Cells to initialize their evolution in the *Cell Tracker*. For a particular u-Cell, its initial contour in frame K is obtained by an approach based on the Earth Mover's Morphing (EMM) [32], if it has at least one corresponding cell in the temporal context. If the u-Cell has no corresponding cell and is away from the image border, then it will be directly copied to frame K as the initial contour. (See Appendix 1.3 for the implementation details.) Otherwise, it will be treated as a leaving cell and the associated trajectory will be terminated.

B. Cell Tracker

The *Cell Tracker* module employs an open active contour model to segment and track cells simultaneously. For *M. xanthus*, the cell contours are represented as Ribbon Snakes [29]. Specifically, each contour is of a parametric form $\phi(s) = (x(s), y(s), w)$, where $(x(s), y(s))$ is a parametric curve (representing the *cell centerline*) and w is the half width of the contour (see Fig. 5). The objective is that the parametric curve can converge to the cell centerline. In practice, $\phi(s)$ is represented as an ordered list of control points. The number of control points for each curve is fixed. Since *M. xanthus* in our data may range from 20 to 80 pixels long, 20 control points are used for a balance of simplicity and accuracy. The half width w is computed from the segmentation result of the same cell in the preceding frame and is the same for each control point.

The initial positions of the contours are determined by interpreting the temporal-context extracted by the *Context Builder* (see Section II-A2). Because the motion of an *M. xanthus* cell is led by one of its two poles, the displacements of the poles could be larger than the displacement of the rest of the cell body. To reduce the portion of the initial contour potentially residing outside of the expected cell body, the length of the initial contour will be shortened by η from the two tips before evolution, where η is a value in (0, 1) indicating the curve shortening ratio.

The evolution of each contour ϕ is governed by the following equation. All contours evolve simultaneously, namely, we update the positions of all contours at each time step.

$$\frac{\partial \phi}{\partial t} = \mathbf{F}_{internal} + \omega_1 \mathbf{F}_{image} + \omega_2 \mathbf{F}_{repel} + \omega_3 \mathbf{F}_{length} \quad (3)$$

where ω_1, ω_2, and ω_3 are the weight parameters of the forces, chosen empirically. We use $\omega_1 = \omega_3 = 1$ and $\omega_2 = 0{:}25$.

For each control point $(x(s), y(s))$ of a cell, the image force and repelling force are the sum of the respective forces exerted on the two outer boundaries of the contour in the outward normal directions. In addition, extra forces, including $\mathbf{F}_{length}$, $\mathbf{F}_{image}$, and $\mathbf{F}_{repel}$, are applied in the outward tangential direction on the first and last control points of the curve (see Fig. 6). Furthermore, the internal force is exerted on each point to regulate the smoothness of the curve.

1) Image Force—The image force is designed to attract the curves to the cell centerlines. The computation of the image force is based on the following observation. The cell regions are of slightly higher intensity than the average background intensity. The cell regions and the background are mostly separated by narrow bands of low intensity pixels, but giving possibly weak edge evidence (see Fig. 4). Therefore, the Chan-Vese model is selected for computing the image force [30], instead of edge-based algorithms. Specifically, the magnitude of the image force at a point $\mathbf{p}$ is $IM(\mathbf{p}) = -((I(\mathbf{p}) - \mu_1)^2 - (I(\mathbf{p}) - \mu_2)^2)$, where $I(\mathbf{p})$ is the intensity at $\mathbf{p}$, μ_1 is the average intensity inside the corresponding cell region enclosed by the outer boundary, and μ_2 is the average background intensity and computed by averaging the intensity of pixels not within any region enclosed by evolving contours.

2) Repelling Force—A contour will suffer from the repelling force defined by the *Barrier Map* when it moves close to other contours. Let $\{\phi_1, \phi_2, \ldots, \phi_n\}$ be all evolving contours. Suppose $\mathbf{p}$ is a point on the outer boundary of ϕ_i, or the first or last control point of ϕ_i. Then, the magnitude of the repelling force at $\mathbf{p}$ is $RP(\mathbf{p}) = BM(\mathbf{p}) + EC(\mathbf{p})$, where $BM(\mathbf{p})$ denotes the force from the *Barrier Map*, and $EC(\mathbf{p})$ denotes the force from the *Evolving Contours* other than ϕ_i.

$$BM(\mathbf{p}) = \begin{cases} 1/[1+\exp(2d(\mathbf{p}) - L_{repel})] & :0 \le d(\mathbf{p}) \le L_{repel} \\ 0 & :\text{otherwise} \end{cases}$$

$$EC(\mathbf{p}) = \sum_{j \neq i} F_{ij}(\mathbf{p}), \text{ where}$$

$$F_{ij}(\mathbf{p}) = \begin{cases} 1/[1+\exp(2d_j(\mathbf{p}) - L_{repel})] & :0 \le d_j(\mathbf{p}) \le L_{repel} \\ 0 & :\text{otherwise} \end{cases}$$

Here, $d(\mathbf{p})$ is the shortest distance from $\mathbf{p}$ to any point on the skeleton of any r-Cell; $d_j(\mathbf{p})$ is the distance from $\mathbf{p}$ to ϕ_j; L_{repel} is a constant cut-off value of the furthest distance that the repelling force can act on. As a rule of thumb, L_{repel} can be set as the average cell thickness.

3) Length Force—Between consecutive frames, the shape of a cell should be consistent. As of *M. xanthus*, certain cell lengths can be expected (except for cell divisions). Inspired by [19], we take such information as prior knowledge and impose onto the evolution. Specifically, the initial contours are shortened by a certain ratio in order to reduce the possibility of part of an initial contour residing outside of the cell body. Then, $\mathbf{F}_{length}$ is

imposed on the first and last control points of the curve in the tangential direction outwards to make the cell grow into the expected length.

Suppose the length of a particular u-Cell in frame $K-1$ is L^{K-1}. Then the target length of this cell in frame K, denoted by L_T^K, is set as L^{K-1}. (The special cases for setting the target length are presented in Appendix 1.1.) The magnitude of $\mathbf{F}_{length}$ is defined as $SH_i = \gamma_i \cdot ds$, where $i \in \{1, 2\}$ is the index of the two tips. Suppose the current length of the contour is L.

$$ds = \begin{cases} 1 & :L < 0.85L_T^K \\ 0.5 & :0.85L_T^K \le L < 0.95L_T^K \\ 0 & :0.95L_T^K \le L < 1.2L_T^K \\ -1 & :L \ge 1.2L_T^K \end{cases}$$

The value of γ_i depends on the image force and repelling force. Suppose v_1 and v_2 are the tangential components of $\mathbf{F}_{image} + \mathbf{F}_{repel}$ at the two tips, respectively.

$$\begin{cases} \gamma_1 = \gamma_2 = 1 & :v_1 \ge 0, v_2 \ge 0 \\ \gamma_1 = 0, \gamma_2 = 1 & :v_1 < 0, v_2 \ge 0 \\ \gamma_1 = 1, \gamma_2 = 0 & :v_1 \ge 0, v_2 < 0 \\ \gamma_1 = 0, \gamma_2 = 2 & :v_1 < v_2 < 0 \\ \gamma_1 = 2, \gamma_2 = 0 & :v_2 < v_1 < 0 \end{cases}$$

In general, $v_i > 0$ indicates expansion, and $v_i < 0$ means the contour will be compressed on that tip. Thus, when only one of the two v_i's is positive, the length force will be exerted only on the tip that is not compressed; when both v_i's are negative, the length force will act only on the tip suffering less compression.

4) Internal Force—$\mathbf{F}_{internal}$ 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 weight parameters (we use $\alpha = 0.4$ and $\beta = 0.2$). For dealing with the high order derivatives of the internal force, the implementation details are presented in Appendix 1.2.

The evolution of a particular contour will stop when the contour length is within $L_T^K \pm 1$ and the external force exerted on the contour is no larger than 0.1. The maximum number of iterations for all contours is $Iter_{max}$.

Finally, the *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 a second degree polynomial is performed to smooth the values. If the minimum is located near the middle of the centerline (i.e., away from the middle less than 10% of the cell length) and the ratio of the maximum to the minimum is larger than θ, a threshold determined experimentally, then the contour will be split in the middle. The contours of the daughter cells will shrink by η first, and then evolve (the newly split contours) 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 [16]. Our trial studies showed that our division checking is effective for our problem, and thus is adopted for the sake of simplicity.

We should mention that certain contours may fail to converge to or locate the expected cells, mainly due to severe image cluttering. Two examples are shown in Fig. 7. A particular contour will be removed after evolution if its length is shorter than L_{skip} (a pre-determined value based on prior knowledge) or the average intensity along the final contour differs by more than 50% from that of the initial contour. Again, we adopt a simple heuristic for the sake of simplicity and acceptable effectiveness, even though more advanced postprocessing methods, such as the convergence failure checking in the twin-separation model [33], are available.

III. Experiments and Evaluations

Our approach was implemented using Matlab (Version: 2014a) with the image processing toolbox and optimization toolbox. The complete set of all codes will be available at http://www3.nd.edu/~jchen16/project/cmark.html.

A. Data

We conducted experiments on ten in-house image sequences of *M. xanthus*. The image sizes range from 300 × 500 to 900 × 1200 pixels, while each pixel is about 0.1 μm × 0.1 μm. Briefly, a CTT-agar pad was formed on a microscope slide using silicon isolator gaskets. The bacteria were inoculated on the top of the pad and covered with a cover slip. After a period of 12–32 hours, a movie of 5 second intervals was made using a 60x oil-immersion objective. The specific bacteria strains of *M. xanthus* used in our experiments were DK8621 and DK1622.

B. Pre-segmentation

The pre-segmentation of our method consists of two steps. (i) A pixel-wise classification is performed using [34]. The output of [34] is a probability map, namely, the probability of being on cells is computed for each pixel. A 3 × 3 mean filter is applied to the probability map to smooth the values. Then, a binary image is obtained from the probability map using the Otsu thresholding [35]. (ii) The detected regions are separated at the intersection points of the region centerlines or at each centerline point whose curvature is larger than 0.25, an experimentally determined threshold.

C. First Frame Refinement

As mentioned in Section II, manual correction may be performed on the pre-segmentation of the first frame to serve as additional prior knowledge, which could be helpful in some difficult cases. To fully automate our approach, we propose a method to refine the pre-segmentation of the first frame automatically. The refinement method inherits the same idea of our hybrid framework, namely, using the spatial-temporal context to guide the active contour evolution, to obtain refined segmentation. Because the spatial-temporal context is built totally from error-prone pre-segmentation, the interpretation of the context may contain

errors in extreme cases. As a result, the proposed automatic refinement may achieve comparable quality as manual correction, but may contain a few errors in some difficult scenarios (the results are reported in Table II). This confirms the objective of manual correction, providing extra prior knowledge to tackle extremely difficult cases.

The refinement approach works as follows. At the beginning, we apply the *Context Builder* module, i.e., the hierarchical association scheme, to the pre-segmentation of the temporal context (frames 1 to 5 in our case). By analyzing the context, we check each cell detection in the first frame and decide whether it needs refinement. We will prepare the initial contours and target lengths (as discussed in Section II-B3) for the cells that need to be refined (by the *Cell Tracker* module), while those "no-further-refinement" detections will constitute the *Barrier Map* (as in Section II-B2).

Two types of detections will remain unchanged, namely, a detection close to the image border or a detection with a unique "child" of a similar length. A detection away from the image border and with no "child" in the context will be discarded (considered as false positive detection).

Except unchanged detections, the initial contours and target lengths will be computed, with two cases. (1) If multiple detections, say $\{P_1, P_2, \ldots, P_k\}$, in the first frame have a common "child", say Q, and Q has a unique path of descendants of similar lengths in the subsequent frames, then the centerline of the longest length among $\{P_1, P_2, \ldots, P_k\}$ will be used as the initial contour (discarding the remaining detections), and the target length will be set as the maximum length of Q and its descendants. (2) Otherwise, a detection, say P, in the first frame will adopt its centerline as the initial contour. The target length will be set as the maximum value among the length of P and the lengths of P's descendants down to the last frame of the context or before the frame in which the corresponding descendant of P has multiple parents.

The first case above is intended to handle false split detections in the first frame, while the second case aims to refine partial segmentations. Note that the *Cell Tracker* will perform cell division checking, in order to handle false merge detections. Moreover, false negative detections, if any, in the first frame do not need special treatment. They will be treated as c-Cells (as shown in Section II-A) in the hybrid framework and start to be kept track of as soon as the appearance is confirmed.

In our experiments, the automatic refinement results contain errors only in two (out of 10) movies. The segmentation and tracking results using the automatic refinement and manual correction are both presented in Table II. It is evident that our fully automated approach can achieve comparable accuracy as the semi-automated version, while human intervention may provide extra assistance to deal with the highly difficult situations.

D. Selection of Parameters

Table I gives a list of the parameters in our hybrid framework that may need adjustments in different datasets. Most of the parameters can be determined according to the images to be analyzed. For example, L_{repel} in the definition of the repelling force is roughly set as the

average cell thickness in the images. On the other hand, there are some parameters in the *Cell Tracker* that may not be derived straightforwardly from the data, and are related to the performance of contour evolution. Hence, it is important to demonstrate that such parameters are not sensitive to our selection.

The parameters to be evaluated are ω_1, ω_2, ω_3 (the weights of the image force, repelling force, and length force in Equ. (3), respectively), α, β (the weights of the two terms in the internal force), N (the number of control points of each contour), and γ (the time step of the contour evolution, see Appendix 1.2). We randomly select five movies and manually segment the first two frames in each of them. All the cells in the first frame are directly propagated as the initial contours for the second frame. After the evolution terminates, we compare the evolution results against the manual segmentation results.

Metric: The metric is adapted from the F1-score and adjusted particularly for *M. xanthus*. The basic idea is to measure the difference in length between the segmented cells and the ground truth. Suppose there are n true cells in the manual segmentation, denoted by $\{G_1, G_2, \ldots, G_n\}$, and m segmented cells, denoted by $\{S_1, S_2, \ldots, S_m\}$. A segmented cell Si is a good segmentation of G_j, if $||S_i \cap G_j|| > 0.5 \cdot ||G_j||$ and $||S_i \cap G_j|| > 0.5 \cdot ||S_i||$. It guarantees that each segmented cell is a good segmentation of at most one ground truth cell, and there is at most one good segmentation for each ground truth cell. Then, the centerlines of S_i and G_j are extracted (using morphological thinning), denoted by Sc_i and Gc_j, respectively. For each pixel of Sc_i, if it does not overlap with G_j, then it is counted as a False Positive (FP) pixel. For each pixel of Gc_j, if it does not overlap with S_i, then it is counted as a False Negative (FN) pixel. In addition, the pixels of the centerlines of all segmented cells that are not a good segmentation of any ground truth cell are considered as FP pixels; the pixels of the centerlines of all ground truth cells without good segmentation are considered as FN pixels. Suppose the total number of pixels of the centerlines of all segmented cells is n_S, and the total number of pixels of the centerlines of all ground truth cells is n_G. Then, the Precision is $Pr = (n_S - FP)/n_S$, and the Recall is $Re = (n_G - FN)/n_G$. Finally, the accuracy is measured by the F1-score, i.e., $F = 2 * Pr * Re/(Pr + Re)$.

We choose to evaluate the results by pixels on the center-lines, instead of pixels in the whole cell regions, because the cell boundaries are not well-defined. The manually segmented thickness of the same cell may vary from person to person, and may even be inconsistent for the same person at different times. Due to the specific shape of *M. xanthus*, there is a smaller impact from the subjectiveness on the cell length than the width. Moreover, we did not adopt the measure in [14], which we found may underestimate some particular errors, such as partial segmentation (i.e., not wholly segmented). For instance, a segmented cell with a length 1/3 shorter than the actual length may still be taken as a good segmentation with no error. Thus, we proposed the above metric.

The results are shown in Fig. 8. The objective of these experiments is not to select the parameters for achieving the highest accuracy (which may cause overfitting). Instead, these experiments are intended to demonstrate that the selected parameters are within the range that the overall accuracy is stable and high. In other words, slight change in any of the selected parameter values will not degrade the performance.

E. Tracking Performance

The metric for the tracking performance is the number of *complete trajectories*. This is a harsh metric because merely a few cell correspondences in fault may cause a huge loss in the number of complete cell trajectories. All the results were examined visually by human experts.

Our method is compared against four state-of-the-art algorithms [10], [14], [16], [19]. The method in [19] is a representative one for the model evolution framework. The other three are different detection association approaches: [10] performs frame-wise optimal matching, [16] conducts global optimization across the whole image sequence, and [14] is a hierarchical scheme based on the EMD matching model.

For a fair comparison, the same pre-segmentation was used for our method and for [14] and [10]. The results of [16] were obtained by the off-the-shelf software, ilastik (http://ilastik.org/), while the parameters were determined by following the software manual.

The method in [19] cannot be reimplemented exactly, due to insufficient implementation details available. We tried our best to make it perform as good as possible, following common practice. In addition, our method needs no human intervention. But, the model evolution method [19] may need other human interventions, as discussed in [19].

The implementation of [10] strictly followed the description in [10]. The only difference is the matching cost. The original definition was for small disk-shape cells. For a fair comparison, we used the same cost as in [14], which was specifically designed for elongated cells. As for [14], the implementation and all the parameters are exactly the same as in [14].

All tracking results are shown in Table II. In dataset 2, 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 [19] in dealing with large cell displacements. Also, the results showed that our method has better performance than the three detection association approaches.

F. Segmentation Performance

We compared the segmentation results with two segmentation methods to show the benefit of leveraging temporal context in segmentation. First, the Otsu thresholding [35] is used as the baseline method. Second, ilastik [34] is one of the best open source software for cell segmentation based on pixel classification. Except ours, the other methods are all performed on individual image frames.

We manually segmented 20 frames selected randomly in each dataset as the ground truth. The metric used to evaluate segmentation accuracy is the same as the F1-score defined in Section III-D. The results in Table II show that our hybrid framework achieves higher segmentation accuracy than the methods which do not utilize the temporal context. In addition, the pre-segmentation accuracy is reported to show how much the *Cell Tracker* can help in improving the pre-segmentation results. For example, in dataset 7, the accuracy of final segmentation is very close to that of pre-segmentation. Meanwhile, the tracking

performance of our hybrid scheme is somewhat similar to [14]. Recall that our *Context Builder* module stems from [14]. Actually, most of the pre-segmentation are confirmed as good segmentation by the temporal context, and the *Cell Tracker* module may not be invoked frequently to refine the results. It shows the flexibility of our hybrid framework.

For low quality images, such as dataset 6, the pre-segmentation is refined a lot. On the other hand, there are also some situations, like dataset 4, where the improvement is not prominent. This is because some image frames are so blurred that either quite some cell tracks are lost or the whole cell bodies are hard to extract.

IV. Discussion and Conclusions

In a nutshell, we present a hybrid framework for segmentation and tracking of *M. xanthus* in time-lapse images, and an open active contour model specially for *M. xanthus* which can be seamlessly integrated with our proposed framework. Our approach systematically combines two major powerful frameworks, the detection association framework and model evolution framework. Indeed, we carefully design the hybrid framework such that our detection association method (i.e., the *Context Builder* module) is used to build preliminary correspondences among cells across different frames, and our model evolution method (i.e., the *Cell Tracker* module) is invoked wherever the preliminary correspondences are suspicious. Our hybrid framework inherits the flexibility in handling various cell behaviors and the robustness to segmentation errors from detection association methods, and also attains the superiority of model evolution methods in leveraging temporal context information. The evaluation shows that our approach outperforms state-of-the-art cell tracking algorithms and effectively utilizes temporal information to achieve higher segmentation accuracy.

The current framework still has two major limitations. First, in the *Context Builder* module, an EMD matching model is used to establish the temporal context. The model has a size-consistency assumption, namely, the size of a cell does not have large change in consecutive frames, and a cell undergoing division is split in the middle so that the sum of the sizes of the two daughter cells is almost equal to that of the parent cell. This is true for certain types of cells, such as *Myxococcus xanthus* and *Pseudomonas aeruginosa*, but not for all cell types. Second, the *Cell Tracker* module aims to leverage the spatial-temporal context to segment and track cells, wherever it is hard to correctly detect them within a single frame. In general, the *Cell Tracker* can refine the pre-segmentation results in low-quality images. But, when an image is heavily blurred, even only in a local area, the *Cell Tracker* may still fail since too little valid information on cells in such a local area is available.

As one of our future works, we plan to apply our approach to different types of cells in both 2D and 3D images. For a specific cell type, the ground distance in the EMD matching model may need to be designed accordingly to capture the specific domain knowledge so that the temporal context can be established appropriately. In addition, different model evolution methods may be needed for different applications, such as 3D active meshes [18] for 3D+t tracking problems.

Supplementary Material

Refer to Web version on PubMed Central for supplementary material.

Acknowledgments

This work was supported in part by NSF Grant CCF-1217906 and NIH Grants 1R01-GM095959 and 1R01-GM100470. The authors would like to acknowledge Cameron Harvey and Shant Mahserejian for preparing the experimental datasets.

References

1. Meijering E, Dzyubachyk O, Smal I. Methods for cell and particle tracking. Methods Enzymol. 2012; 504(9):183–200. [PubMed: 22264535]
2. Padfield D, Rittscher J, Roysam B. Quantitative biological studies enabled by robust cell tracking. ISBI. 2011:1929–1934.
3. Lienkamp SS, Liu K, Karner CM, Carroll TJ, Ronneberger O, Wallingford JB, Walz G. Vertebrate kidney tubules elongate using a planar cell polarity-dependent, rosette-based mechanism of convergent extension. Nature Genetics. 2012; 44(12):1382–1387. [PubMed: 23143599]
4. Liu K, Lienkamp SS, Shindo A, Wallingford JB, Walz G, Ronneberger O. Optical flow guided cell segmentation and tracking in developing tissue. ISBI. 2014:298–301.
5. Harvey CW, Morcos F, Sweet CR, Kaiser D, Chatterjee S, Liu X, Chen DZ, Alber M. Study of elastic collisions of Myxococcus xanthus in swarms. Physical Biology. 2011; 8(2):026016. [PubMed: 21471636]
6. Balagam R, Litwin DB, Czerwinski F, Sun M, Kaplan HB, Shaevitz JW, Igoshin OA. Myxococcus xanthus gliding motors are elastically coupled to the substrate as predicted by the focal adhesion model of gliding motility. PLoS Computational Biology. 2014; 10(5):e1003619. [PubMed: 24810164]
7. Gibiansky ML, Hu W, Dahmen KA, Shi W, Wong GC. Earthquake-like dynamics in Myxococcus xanthus social motility. Proceedings of the National Academy of Sciences. 2013; 110(6):2330–2335.
8. Zhang Y, Ducret A, Shaevitz J, Mignot T. From individual cell motility to collective behaviors: Insights from a prokaryote, Myxococcus xanthus. FEMS Microbiology Reviews. 2012; 36(1):149–164. [PubMed: 22091711]
9. Liu X, Harvey C, Wang H, Alber M, Chen D. Detecting and tracking motion of Myxococcus xanthus bacteria in swarms. MICCAI 2012. Part I. 2012:373–380.
10. Padfield D, Rittscher J, Roysam B. Coupled minimum-cost flow cell tracking for high-throughput quantitative analysis. Medical Image Analysis. 2011; 15(4):650–668. [PubMed: 20864383]
11. Li F, Zhou X, Ma J, Wong ST. Multiple nuclei tracking using integer programming for quantitative cancer cell cycle analysis. IEEE Trans on Medical Imaging. 2010; 29(1):96–105. [PubMed: 19643704]
12. Magnusson KE, Jaldén J, Gilbert PM, Blau HM. Global linking of cell tracks using the Viterbi algorithm. IEEE Trans on Medical Imaging. 2015; 34(4):911–929. [PubMed: 25415983]
13. Xie J, Khan S, Shah M. Automatic tracking of Escherichia coli in phase-contrast microscopy video. IEEE Trans on Biomedical Engineering. 2009; 56(2):390–399.
14. Chen J, Harvey CW, Alber M, Chen DZ. A matching model based on earth mover's distance for tracking Myxococcus xanthus. MICCAI 2014. Part II. 2014:113–120.
15. Schiegg M, Hanslovsky P, Haubold C, Koethe U, Hufnagel L, Hamprecht FA. Graphical model for joint segmentation and tracking of multiple dividing cells. Bioinformatics. 2014; 31(6):948–956. [PubMed: 25406328]
16. Schiegg M, Hanslovsky P, Kausler BX, Hufnagel L, Hamprecht F. Conservation tracking. ACCV. 2013:2928–2935.

17. Debeir O, Ham PV, Kiss R, Decaestecker C. Tracking of migrating cells under phase-contrast video microscopy with combined mean-shift processes. IEEE Trans on Medical Imaging. 2005; 24(6):697–711. [PubMed: 15957594]
18. Dufour A, Thibeaux R, Labruyere E, Guillén N, Olivo-Marin J-C. 3-D active meshes: Fast discrete deformable models for cell tracking in 3-D time-lapse microscopy. IEEE Trans on Image Processing. 2011; 20(7):1925–1937.
19. Deng Y, Coen P, Sun M, Shaevitz JW. Efficient multiple object tracking using mutually repulsive active membranes. PloS One. 2013; 8(6):e65769. [PubMed: 23799046]
20. Dzyubachyk O, Cappellen WV, Essers J, Niessen WJ, Meijering E. Advanced level-set-based cell tracking in time-lapse fluorescence microscopy. IEEE Trans on Medical Imaging. 2010; 29(3): 852–867. [PubMed: 20199920]
21. Ersoy I, Bunyak F, Higgins JM, Palaniappan K. Coupled edge profile active contours for red blood cell flow analysis. ISBI. 2012:748–751.
22. Ray N, Acton ST, Ley K. Tracking leukocytes in vivo with shape and size constrained active contours. IEEE Trans on Medical Imaging. 2002; 21(10):1222–1235. [PubMed: 12585704]
23. Zimmer C, Labruyere E, Meas-Yedid V, Guillen N, Olivo-Marin J-C. Segmentation and tracking of migrating cells in videomicroscopy with parametric active contours: A tool for cell-based drug testing. IEEE Trans on Medical Imaging. 2002; 21(10):1212–1221. [PubMed: 12585703]
24. Cheng J, Xiong W, Gu Y, Chia S-C, Wang Y, Lim J-H. A three-color coupled level-set algorithm for simultaneous multiple cell segmentation and tracking. ACCV. 2015:268–283.
25. Maška M, Daněk O, Garasa S, Rouzaut A, Muñoz-Barrutia A, de Solorzano CO. Segmentation and shape tracking of whole fluorescent cells based on the Chan–Vese model. IEEE Trans on Medical Imaging. 2013; 32(6):995–1006. [PubMed: 23372077]
26. Bensch R, Ronneberger O. Cell segmentation and tracking in phase contrast images using graph cut with asymmetric boundary costs. ISBI. 2015:1220–1223.
27. Li K, Miller ED, Chen M, Kanade T, Weiss LE, Campbell PG. Cell population tracking and lineage construction with spatiotemporal context. Medical Image Analysis. 2008; 12(5):546–566. [PubMed: 18656418]
28. Chen J, Mahserejian S, Alber MS, Chen DZ. A hybrid approach for segmentation and tracking of Myxococcus xanthus swarms. MICCAI 2015, Part III. 2015:284–291.
29. Mayer H, Laptev I, Baumgartner A. Multi-scale and snakes for automatic road extraction. ECCV. 1998:720–733.
30. Chan TF, Vese LA. Active contours without edges. IEEE Trans Image Processing. 2001; 10(2): 266–277.
31. Rubner Y, Tomasi C, Guibas LJ. The earth mover’s distance as a metric for image retrieval. International Journal of Computer Vision. 2000; 40(2):99–121.
32. Makihara Y, Yagi Y. Earth mover’s morphing: Topology-free shape morphing using cluster-based EMD flows. ACCV. 2011:202–215.
33. Al-Diri B, Hunter A, Steel D. An active contour model for segmenting and measuring retinal vessels. IEEE Trans Medical Imaging. 2009; 28(9):1488–1497. [PubMed: 19336294]
34. Sommer C, Straehle C, Kothe U, Hamprecht FA. ilastik: Interactive learning and segmentation toolkit. ISBI. 2011:230–233.
35. Otsu N. A threshold selection method from gray-level histograms. Automatica. 1975; 11(285–296):23–27.

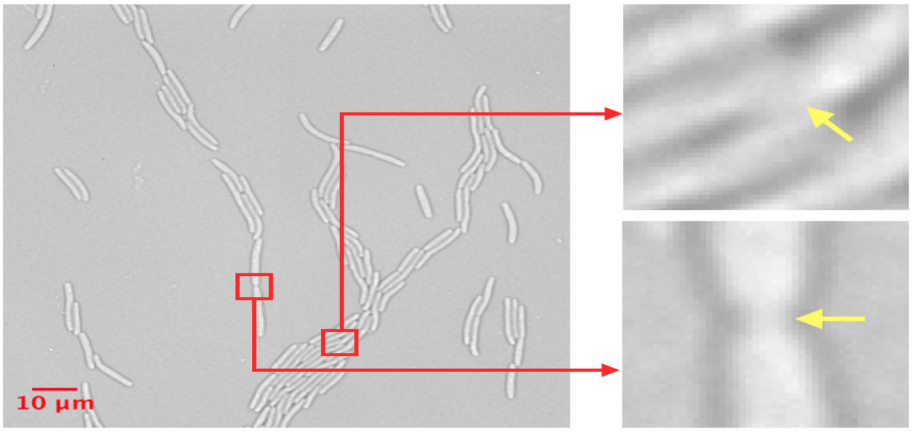

Fig. 1.

A frame in an *M. xanthus* movie. Two zoom-in regions amplify some details of obscure cell boundaries due to touching in a cluster (upper) and head-to-head touching (lower).

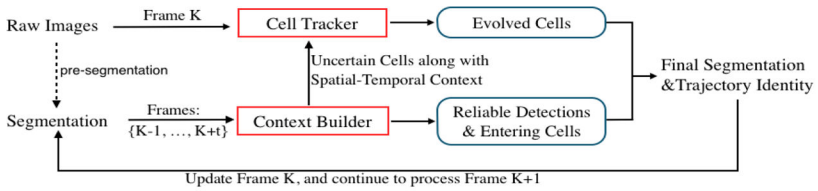

Fig. 2.

The outline of our framework. Two key modules are labeled in red. The dashed line means the pre-segmentation is obtained for each frame first. The solid lines show the work flow to process each frame $K > 1$. See Section III-C for the processing of the first frame. Assume cell trajectories have been built until frame $K - 1$. For frame K, *Context Builder* extracts the spatial-temporal context. Reliable cell detections in frame K are associated with existing trajectories, and reliable entering cells are confirmed. Next, trajectories with uncertain correspondence, together with the spatial-temporal context, are fed to *Cell Tracker* to find their positions in frame K by contour evolution. The reliable cells, entering cells, and evolved cells are put together to produce the final segmentation and trajectory identity of each cell in frame K.

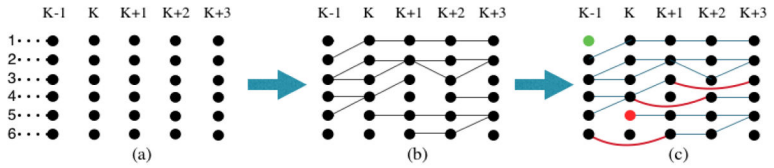

Fig. 3.

Illustration of building the temporal context by hierarchical association. (a) To process frame K, the association scheme is applied to frames $K-1$, …, $K+3$ (the temporal context depth is 5 in this example). Each detected object is represented as a vertex in a graph of five columns, one column per frame. Suppose 6 trajectories, labeled as 1 to 6, have been built up to frame $K-1$. (b) In the low-level association, the correspondence is first built between every two consecutive frames. (c) In the high-level association, the EMD-based matching model is applied to all child-deficient and parent-deficient vertices (see Section II-A for the definitions) to link these vertices within the entire temporal context. **Context Analysis**: Let $V_{i,j}$ denote the cell corresponding to the j-th vertex (from top to bottom) in the i-th column. $V_{K,1}$ can be confirmed as a reliable segmentation and is linked to trajectory 2 if $V_{K-1,2}$ and $V_{K,1}$ have consistent shapes. $V_{K,5}$ can be confirmed as an entering cell and initiates a new trajectory if $V_{K+1,5}$, $V_{K+2,5}$, and $V_{K+3,5}$ have similar shapes as $V_{K,5}$. Trajectory 1 will be terminated if $V_{K-1,1}$ is close to the image border. The remaining trajectories, 3, 4, 5, and 6, will continue by propagating $V_{K-1,3}$, $V_{K-1,4}$, $V_{K-1,5}$, and $V_{K-1,6}$ to frame K, respectively. The **Cell Tracker** will find their corresponding cell regions in frame K by the evolution of active contours. The initial contours will be interpreted from the context. For example, the initial position of trajectory 6 is set as the interpolated position of morphing from $V_{K-1,6}$ to $V_{K+1,6}$.

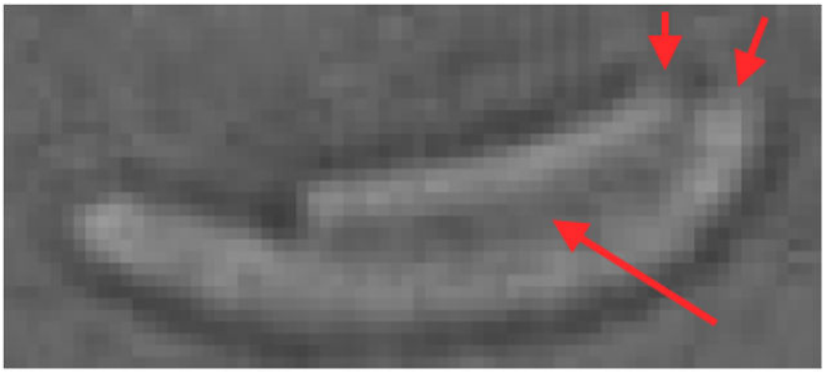

Fig. 4.
An example of cells showing weak edge evidence between the cell bodies and image background.

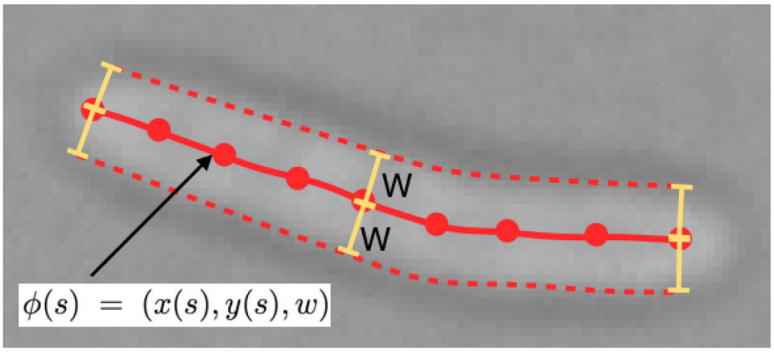

Fig. 5.

Illustrating the Ribbon Snake representation: $\phi(s) = (x(s), y(s), w)$. The parametric curve $(x(s), y(s))$ is represented as an ordered list of control points. The half width w of the contour is the same for each control point.

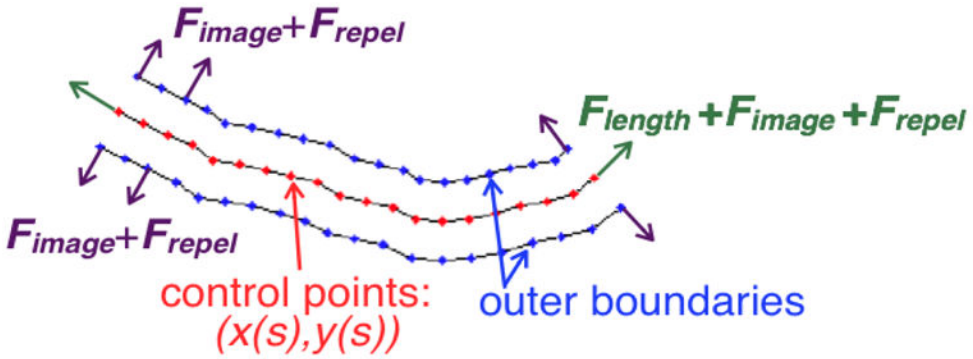

Fig. 6.

Illustrating the forces exerted on an evolving contour. The length force is applied only to the first and last control points in the outward tangential direction. In the normal direction, the forces are the sum of the forces applied to the corresponding outer boundary points.

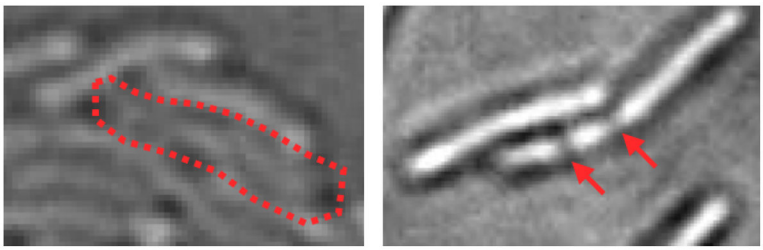

Fig. 7.

Two extreme examples on which the contour evolution may fail. (Left) The appearance of the cell circled by red is largely degraded due to tightly packed neighboring cells. As a result, the evolving contour may not converge to the expected cell centerline. (Right) The cluttered background alters the intensity along the cell body (indicated by red arrows), thus possibly causing broken contours or leakage.

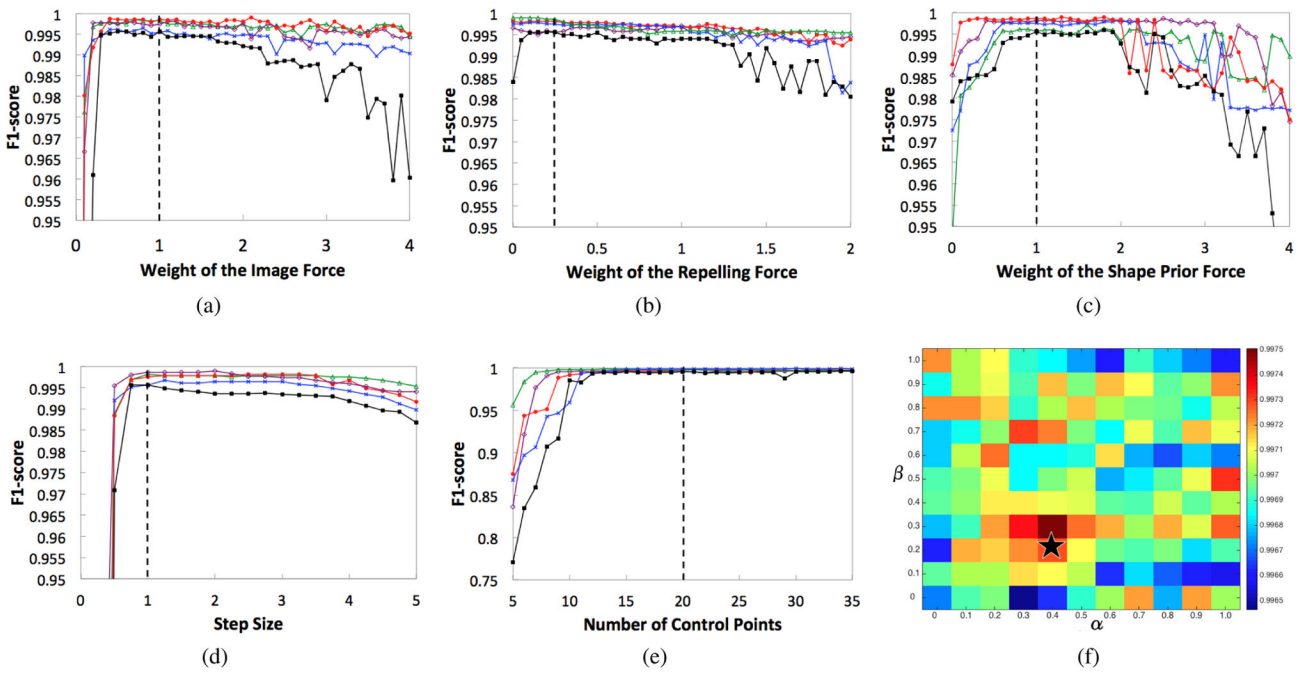

Fig. 8.

Evaluating the stability of the parameters for controlling the contour evolution. (a)–(e) The performance in five experiments of varying the values of the parameters, ω_1 (the weight of the image force), ω_2 (the weight of the repelling force), ω_3 (the weight of the length force), γ (the evolution step size), and N (the number of control points to represent a cell curve). The colored curves in the figures represent the results of different experiments. The values used to produce the results in Table II are indicated by the dashed lines. (f) The performance of varying the weight parameters of the two terms in the internal force, α and β. The plot is averaged over all five experiments. The selected values are indicated by the star.

TABLE I

The parameters of our proposed framework that may need adjustment for different datasets.

Category	Notation	Description	Default Value	Remark
Context Builder	t	t + 2 is the depth of the temporal context	3	Commonly used values: 1, 2, 3
	ρ	The minimum acceptable ratio between the flow amount and cell size in low-level association	0.75	Reasonable range: (0.6, 0.9)
	λ	The maximum acceptable matching cost of correspondence in low-level association	2	Decrease for more conservative low-level association
Cell Tracker	η	The shortening ratio of an initial contour	0.2	May reduce for more rigid bacteria
	$Iter_{max}$	The maximum number of iterations used for contour evolution	60	80% of the estimated average number of pixels along cell centerlines
	L_{skip}	The length threshold to discard short objects	10	Max(10, 0.9*$shortest\ cell\ length$)
	L_{seg}	The maximum discrepancy in length for reliable segmentation	6	Estimated average cell thickness
	L_{enter}	The maximum discrepancy in length for entering cells	10	Increase when cells may move very fast
	θ	The threshold for determining potential cell division	2	Increase when the cell boundaries exhibit stronger edge evidence in general
	L_{repel}	The cut-off distance for repelling force to take effect	6	Estimated average cell thickness

TABLE II

Evaluation and comparison of tracking and segmentation performance on 10 different movies. The difficulty in dataset 2 is due to large cell displacement in consecutive frames. The difficulty in datasets 4 – 6 are resulted from degraded image quality. The major difficulty in datasets 7 – 10 is the tight head-to-head cell touching. The results of our approach without manual correction, i.e., using automatic refinement, are reported inside parentheses (which are shown only when the automatically refined segmentation in the first frame still contains errors).

Datasets				**Tracking Results**					**Segmentation Accuracy**			
	# of Frames	**# of Trajectories**	**Difficulty**	**Ours**	**[19]**	**[14]**	**[10]**	**[16]**	**Ours**	**Otsu**	**[34]**	**Pre-Seg**
1	46	43	Low	**41**	25	39	23	27	**0.9677**	0.8174	0.9016	0.9503
2	47	74	High	**68** (64)	5	59	32	21	**0.9842** (0.9775)	0.6872	0.9361	0.9713
3	60	48	Low	**47**	37	44	44	38	**0.9919**	0.7422	0.9659	0.9841
4	69	50	High	**45** (44)	21	31	19	19	**0.9340** (0.9320)	0.8151	0.8000	0.9104
5	71	79	High	**70**	18	64	46	61	**0.9726**	0.6991	0.6616	0.9648
6	81	52	High	**51**	26	42	30	25	**0.9865**	0.5691	0.8241	0.9390
7	85	108	Medium	**105**	80	103	82	84	**0.9868**	0.7928	0.9486	0.9857
8	101	116	Medium	**107**	75	100	91	84	**0.9670**	0.8064	0.9483	0.9486
9	120	113	Medium	**111**	76	100	90	94	**0.9912**	0.8480	0.9559	0.9785
10	331	164	Medium	**151**	57	115	78	85	**0.9857**	0.8134	0.9461	0.9737