

Segmentation and Track-analysis in Time-lapse Imaging of Bacteria

Sajith Kecheril Sadanandan*, Özden Baltekin, Klas E. G. Magnusson, *Student member, IEEE*, Alexis Boucharin, Petter Ranefall, Joakim Jaldén, *Senior member, IEEE*, Johan Elf, Carolina Wählby, *Senior member, IEEE*,

Abstract—In this work, we have developed tools to analyze prokaryotic cells growing in monolayers in a microfluidic device. Individual bacterial cells are identified using a novel curvature based approach and tracked over time for several generations. The resulting tracks are thereafter assessed and filtered based on track quality for subsequent analysis of bacterial growth rates. The proposed method performs comparable to the state-of-the-art methods for segmenting phase contrast and fluorescent images, and we show a 10-fold increase in analysis speed.

Index Terms—*E. coli*, segmentation, time-lapse, microscopy, tracking.

I. INTRODUCTION

LIVE cell experiments on prokaryotic cells pave the way to understand the complex biological functions of living organisms. Many live cell experiments require monitoring of cells under different conditions over several generations. Isogenic cells display cell-to-cell variability even when grown under similar conditions [1]. To study the origin and consequences of such variation it is necessary to monitor many individual cells for extended periods of time to reach statistically testable conclusions [2]. Time-lapse experiments usually generate large quantities of data, which become extremely difficult for human observers to evaluate in an unbiased way [3]. Thus, automated systems are necessary to analyze such datasets in order to reach robust and reproducible results.

Time-lapse imaging of growing bacterial cells are important both to answer fundamental biological questions related to the bacterial cell cycle as well as to study responses to changes in growth conditions due to changes in nutrients or antibiotics [4]. Based on the growth conditions and imaging modalities, various automated image segmentation and tracking packages were developed. For example: *MicrobeTracker* [5] was designed to segment phase contrast images and detect fluorescent spots in a parallel fluorescent channel in bacterial cells grown isolated or in micro-colonies on agarose pads; *Schnitzcells* [6]

was specifically designed to analyze fluorescent time-lapse images of *E. coli* (*Escherichia coli*) grown on agarose; and *MAMLE* [7] was designed to analyze *E. coli* from phase contrast and fluorescent images.

Most image segmentation methods rely on raw pixel intensities to get an initial segmentation result, and further refined segmentation depends on this initial segmentation. *MicrobeTracker* finds an initial segmentation using Otsu's thresholding method [8] followed by edge detection and watershed segmentation. In the final step, active contours are applied to refine object boundaries. *MicrobeTracker* needs manual correction of the first frame to get satisfactory segmentation results for time-lapse images. It is difficult to analyze large image sequences in *MicrobeTracker* due to its inherent memory problems, and it is often necessary to modify the code for practical applications. *MAMLE* uses range filtering to find an initial segmentation result, followed by multi-scale edge detection and a maximum likelihood classification to correct over- and under- segmentation. In *Schnitzcells*, initial segmentation is achieved by edge detection followed by post processing to correct segmentation errors.

Phase contrast images of *E. coli* exhibit high-intensity regions inside cellular regions comparable to, or even brighter than, regions between cells. Relying on raw intensity therefore leads to over- and under-segmentation at the same time. The problem is amplified when trying to track cells over time. Even a 1% error in detection of cells in every frame renders a cell lineage useless for further analysis if many cells are tracked over a long time.

Previously published cell tracking algorithms rely on model evolution, where a model of the cell is evolved over time using techniques such as active contour models [9] or level sets [10], and tracking by detection [11]. Tracking by detection involves two stages; segmentation and tracking. Sometimes both these steps are combined together to get a final tracking result [12].

In this work, we use tracking by detection, i.e., we separate the segmentation and tracking problems and solve them separately, followed by a quality control and refinement step where some of the segmentation and tracking errors are corrected. Cell segmentation is done using our novel Curvature Based Approach, hereafter called as CBA, and tracking is done using a state-of-the-art tracking algorithm [13]. In the following sections we present our segmentation methodology and compare it with that of *MicrobeTracker* and *MAMLE* on phase contrast as well as fluorescence images, from our own and previously published experiments. In this work, we focus on cell tracking of phase contrast microscopy images.

S. K. Sadanandan, P. Ranefall and C. Wählby are with the Department of Information Technology, Uppsala University, Sweden and SciLifeLab, Sweden. C. Wählby is also with Broad Institute of Harvard and MIT, Cambridge, MA, USA. e-mail: sajith.ks@it.uu.se

A. Boucharin, Ö. Baltekin and J. Elf are with the Department of Cell and Molecular Biology, Computational and Systems Biology, Uppsala University, Sweden

K. E. G. Magnusson and J. Jaldén are with ACCESS Linnaeus Centre, KTH Royal Institute of Technology, Stockholm, Sweden

Copyright (c) 2015 IEEE. Personal use of this material is permitted. However, permission to use this material for any other purposes must be obtained from the IEEE by sending a request to pubs-permissions@ieee.org

After segmentation, we track the cells through the time-lapse sequence and perform post tracking segmentation correction to get a final segmentation result. Finally, we show how the combined segmentation and tracking approach can be applied to quantify differences in cell growth rate under different experimental conditions.

II. METHODOLOGY

A. Strains, Media and Growth Conditions

For the experiments with fluorescence detection, the *E. coli* strain expressed the red fluorescent protein turbo RNA from a constitutive chromosomal promoter (BW25993 intC::P2rrnB-tRFP KanR) and cells are grown in LB-media supplemented with surfactant (Pluronic® F-108, Sigma #542342, 0.85% (w/v) final concentration). For the phase contrast experiments, we used *E. coli* strain with chromosomal fusion of yellow fluorescent protein Venus to the lactose permease (SX700 lacY-venus, as described in Choi et al [14]). The cells are grown in M9 defined minimal media supplemented with surfactant (Pluronic® F-108, Sigma #542342, 0.85% (w/v) final concentration), Amino Acids (RPMI 1640 Amino Acids Solution (50×), Sigma # R7131) and a carbon source. The carbon source can be Glycerol (43 mM) or Glucose (22 mM).

The bacterial cell colonies in our own experiments were grown on a specially designed microfluidic device made from Polydimethylsiloxane (PDMS) [15][16] with 51 parallel growth chambers (traps) of size $40 \times 40 \times 0.9 \mu\text{m}$, which are open to growth media exchange at two ends. The cells grow in a single layer and as the size of the microcolony gets larger than the trap, excess cells leave the trap through the outlets, thus maintaining the colony size of approximately 200 cells throughout the experiment. The growth conditions are maintained by pressure driven flow of media, which is controlled by an electronic pressure regulator (Elveflow OB1).

B. Image Acquisition

Phase contrast images were acquired at 125 ms exposure time using a Nikon Ti-E external phase contrast microscope equipped with a CFW-1312M camera (Scion Corporation) and a Nikon Apo TIRF 100× NA 1.49 oil objective. The fluorescence images were acquired using an EMCCD camera (iXon, Andor Technologies), excitation from a 561 nm cw solid state laser (Genesis MX 561-500 STM Coherent) with a ZT561rdc dichroic mirror (Chroma) and two emission filter (585/40 and 561-notch filter), and a Nikon Plan Apo VC 100× NA 1.40 oil objective. The microscope was equipped with a TIR based hardware autofocus that keeps the cells in focus over days. Time-lapse images for phase contrast were acquired at regular intervals 10 s to 30 s (depending on experiment) for fluorescence images they were spaced out to 60 s to limit photo bleaching and laser damage. Bacterial cell colonies from the *MicrobeTracker* [5] and *Schnitzcells* [6] datasets were used for comparison. The datasets used in this work are shown in Fig. 1.

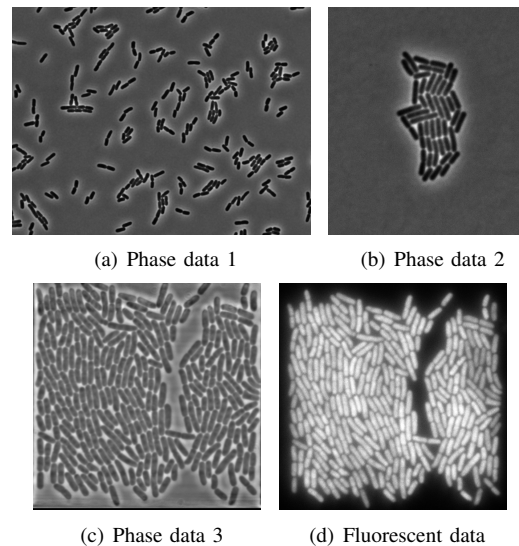


Fig. 1. Input images: a) and b) are images of cells grown on agarose pads and c) and d) are cells grown on PDMS microfluidic device.

C. Image Preprocessing

The input phase contrast images are of size 1360×1024 pixels and contain the cell colony as well as some regions of the microfluidic device. We aligned the image sequences based on image cross correlation to account for the stage repositioning inaccuracy that occurred when cycling through different traps during the image acquisition process. The aligned images were manually cropped to the region containing cells. Since the stack was aligned, manual selection on the first frame was enough to crop the entire image stack. For the fluorescent dataset, since the cells are brighter than background, we inverted the input raw image and performed all steps as described below.

D. Curvature Based Contrast Enhancement

In the input phase contrast images, the *E. coli* cells appear as dark rod-shaped objects on a brighter background. The *E. coli* colony is tightly packed so that the intensity values between the cells are often similar to those inside the cells, and it is common to see high intensity regions inside cells, which implies that any purely intensity based approach for segmentation will result in erroneous output. To overcome this, we tried the separation of principal curvatures of the intensity surface [17]. The regions were thereafter separated based on the minimal curvature as described below.

The curvature was found using techniques from differential geometry [17]. Consider a 1D case, where r is a curve as shown in Fig. 2 and p and q are two points on the curve. We know that the gradient of the curve with respect to the arc length gives the tangent, t , at that point, i.e., $t = r'$. The rate of change of tangent direction as we move along the curve is the curvature of the curve, i.e., $t' = kn$, where n is the unit normal vector to the curve, that is perpendicular to the tangent, and k is the curvature. So we have $r'' = kn$. This shows that the curvature at a point is the second derivative of the curve at that point. The sign of the curvature is determined

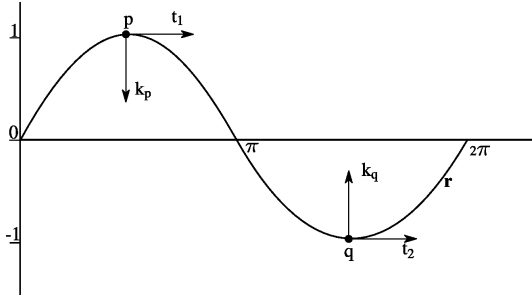


Fig. 2. Principal curvature estimation: k_p is the curvature at point p with negative value and k_q is the curvature at point q with positive value for the curve r .

by whether the slope is increasing or decreasing. Here we can see that it is negative in the maximum point and positive in the minimum point as shown by k_p and k_q (arrow pointing upward as positive and arrow pointing downward as negative). We extended the same idea to the 2D case. Consider a gray scale image as a surface in 3D with (x,y) being the spatial coordinates and $I(x,y)$ being the gray level intensity at that particular spatial location. Following a similar convention as for the 1D case, the image surface is assumed to be continuous with partial derivatives existing at least to order 2 [17]. Here, we first made the image smooth by convolving it with a Gaussian kernel. We set the standard deviation of the Gaussian to 1.4 pixels, which is approximately $1/10^{th}$ the width of the *E. coli* cells, found experimentally.

A particular point on the image surface has an infinite number of curves passing through it. Out of all these curves there are two curves that are particularly interesting. They are the curve with maximum curvature and the curve with minimum curvature, which are orthogonal to each other. These curvatures are equal to the eigenvalues of the Hessian matrix [18]. The Hessian is the second derivative matrix of the image, calculated for every pixel, which is created as

$$H = \begin{bmatrix} I_{xx} & I_{xy} \\ I_{xy} & I_{yy} \end{bmatrix}$$

where I_{xx} and I_{yy} are the 2^{nd} derivatives of the image taken in the x - and y -directions, and I_{xy} is the derivative of the image taken first in the x -direction and then in the y -direction, using discrete approximations [19]. The eigenvalues can be calculated as follows

$$k_{1,2} = \frac{\text{trace}(H) \pm \sqrt{\text{trace}(H)^2 - 4 \times \det(H)}}{2} \quad (1)$$

Here, k_1 and k_2 are the principal curvatures with $k_1 < k_2$. In phase contrast images of *E. coli*, consider two rod shaped cells lying parallel to each other, the cells are dark and the region between the cells is bright. When we calculate principal curvatures in the region between the cells, one is perpendicular to the major axis of the cell and its curvature is negative, and the other one is parallel to the major axis of the cell with zero or small value near zero curvature (positive or negative depending on local intensity values). Taking the lowest value of the two gives the curve with the greatest

curvature magnitude in the negative direction at that point. In this way we can enhance the contrast of the image in bright background regions between cells while avoiding enhancing variations inside the darker cell regions, as shown in Fig. 3.

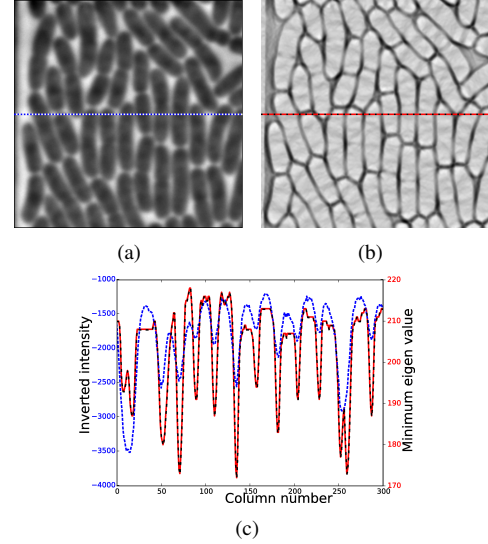


Fig. 3. Curvature-based contrast enhancement: a) Original input image, b) Curvature based contrast enhanced image and c) Plot showing pixel values from the same row from (a) (red line), and (b) (blue line). Note that the plotted pixel values from (a) in (c) are inverted for display.

E. Object Segmentation

We used the presented curvature-based enhancement step to enhance the contrast in the images. Next, we segmented out the cells using a repeated thresholding approach. The contrast-enhanced image is an image with floating point values. In order to make the threshold computation easier, we normalized the image and quantized it to 256 intensity levels. A single threshold value was not sufficient to separate all individual cells, and watershed segmentation resulted in ambiguities in the positioning of the edges of the cells. We therefore used multiple thresholds and prior knowledge about the cell area and the cell shape, in the form of major and minor axes lengths, to filter out the cells from background regions.

For each threshold level, the image was labeled and each object fitted with an ellipse. The ellipse parameters are found using moments [20] as follows. We create a matrix M such that,

$$M = \begin{bmatrix} m_{02} & m_{11} \\ m_{11} & m_{20} \end{bmatrix}$$

$$\text{major axis} = 4 \times \sqrt{\frac{\lambda_1}{m_{00}}} \quad (2)$$

$$\text{minor axis} = 4 \times \sqrt{\frac{\lambda_2}{m_{00}}} \quad (3)$$

λ_1 and λ_2 are eigenvalues of moment matrix M , m_{pq} is the pq^{th} central moment in x and y respectively. m_{00} is 0^{th} central moment (area of the object).

The ellipse parameters of individual objects were analyzed as follows. The objects were filtered based on the major and

minor axis length to remove very large and very small regions. The major and minor axes lengths are given as parameter to the algorithm. For each object, a weight was calculated as follows and assigned to the object.

$$weight = 0.5 \times residual\ area\ ratio + 0.5 \times convexity \quad (4)$$

where the residual area ratio (RAR) is found as

$$RAR = \frac{\min(area, ellipse\ area)}{\max(area, ellipse\ area)} \quad (5)$$

and convexity

$$convexity = \frac{area}{area\ of\ convex\ hull\ of\ object} \quad (6)$$

The ellipse area is found as

$$Ellipse\ area = \frac{\pi \times major\ axis \times minor\ axis}{4} \quad (7)$$

We have already seen that the cellular regions exhibit higher intensity values and edges between cells exhibit lower intensity values on the contrast-enhanced image. Thus we made the assumption that the histogram is bimodal, with the largest peak representing objects and background pixels, while the smaller peak represents edges. The histograms of the contrast enhanced images are shown in Fig. 4. From a computational point of view it is time-consuming to apply a threshold at all the 256 intensity levels and analyze the size and shape of every binary object present in the image. It is also clear that if the threshold is too low then most of the objects merge together and if threshold is too high then the objects are fragmented. So we wanted to limit the threshold levels applied. We experimentally found that searching for a locally optimal threshold within 10 intensity levels around the mean intensity was sufficient to correctly threshold most cellular regions. This value may change with other datasets but was found giving satisfactory results for the 4 datasets used here. This threshold is evaluated later in the paper.

For each threshold, weight values were assigned to every object in the range $[0, 1]$, resulting in a stack of segmentation results. We reduced the stack by a maximum projection, i.e., find the maximum value for every (x, y) coordinate through the stack found, thus keeping the per-object segmentation result with the highest weight. This projection image was thresholded and holes were filled to obtain the final object segments. We found the threshold experimentally and set it to 0.75. The stability of the threshold is evaluated in a later section in this paper. All objects below this threshold were considered to be image background. Fig. 5 shows two examples of weighted thresholded images, max projection of segmentation stack and the hole filled and labeled image. The remaining objects were subjected to further intensity based filtering to remove artifacts and smoothing to remove sharp corners as described below.

F. Object Filtering

Our object filtering is based on the assumption that the intensity distribution of the cells follows a Gaussian distribution,

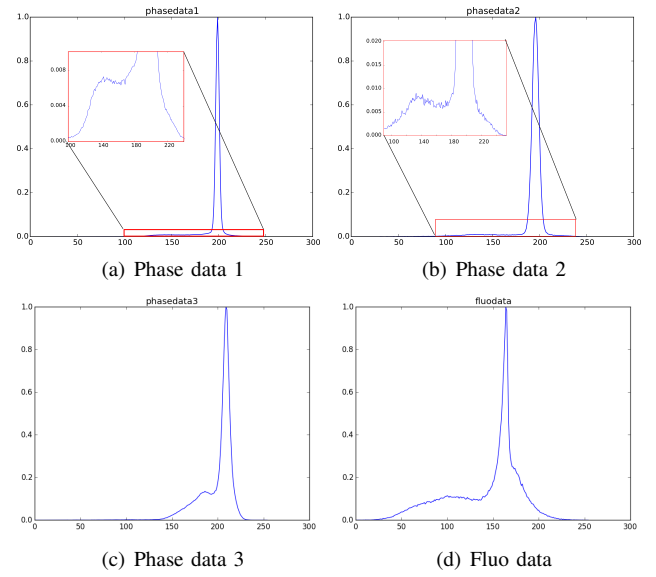


Fig. 4. Histogram of contrast enhanced images: a) and b) show each of the histograms for phase data 1 and phase data 2 with a zoomed-in region to visualize the local minima in the histogram. For c) and d) the bimodal distribution of the histograms can be seen without re-scaling.

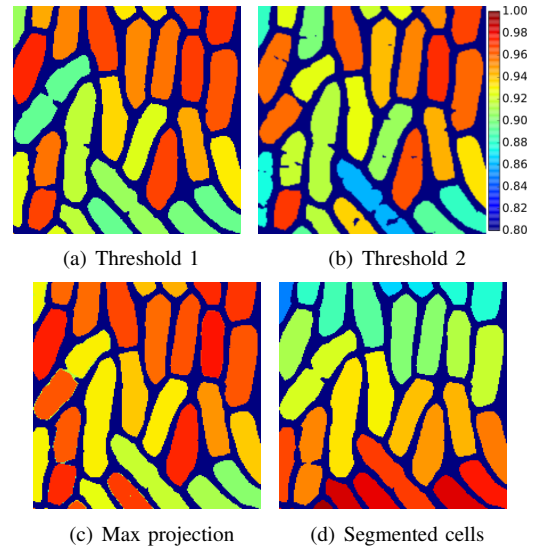


Fig. 5. Intermediate outputs of segmentation: a) and b) show example of weighted images from the segmentation stack, c) Max projected image, d) Output after thresholding and hole filling of (c). In (a), (b) and (c) the colors represent weight values ranging from 0.0 to 1.0 (color map is scaled for illustrations as shown in the color bar) and in (d) color is to distinguish among different segments.

and that outliers are debris. Approximations of the mean and the standard deviation of the distribution were computed as follows. For each of the segmented cells, the median intensity corresponding to the original image is found and stored in a list. The list of intensities was normalized. From the list of normalized intensities, the histogram of intensity is found and interpolated using spline interpolation [21]. On the interpolated distribution, we find the position of the peak, use it as mean, and the half maximum of the peak on both sides of the peak using linear search. We find the full width at half maximum (FWHM) by the difference of the right and left half maximum.

TABLE I
PARAMETERS SET FOR MORPHOLOGICAL SNAKES

Parameter	Value
Alpha	1000
Sigma	4.2
Smoothing	3
Threshold	25
Balloon	-1
Number of iterations	2

We estimated the standard deviation [22] as follows

$$\text{Estimated standard deviation} = \frac{FWHM}{2\sqrt{2\log(2)}} \quad (8)$$

We introduced a threshold, 4 standard deviations above the mean intensity of the obtained distribution. All objects that had values less than this threshold were considered for further processing. We performed the intensity based filtering on all images in the sequence separately to compensate for variations in image intensity.

G. Object Smoothing

Object smoothing is required only for such applications where the segment accuracy is more important than speed of execution. The output objects from the filtered segmentation sometimes have sharp corners. To smooth the borders we applied morphological geodesic active contours [23], which is a fast level set based contour evolution method. In this method, level set evolution is done using dilation and erosion of binary images. We set the parameters as shown in Table I and used the intensity normalized contrast-enhanced image, i.e., the low eigenvalue image, as reference image for the level set evolution. All the parameters have the same meaning as mentioned in [23]. Each segment was smoothed individually to avoid potential merging of segments.

H. Segmentation Evaluation Approach

We evaluated the segmentation performance on a per-cell basis using recall, precision and the harmonic mean of the two, referred to as the F-score on a per object basis. These were found as follows

$$\text{Recall} = \frac{TP}{TP + FN}$$

$$\text{Precision} = \frac{TP}{TP + FP}$$

$$\text{F-score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

where TP is true positive pixels, FP is false positive pixels and FN is false negative pixels.

To have a fair comparison of the evaluated methods, all objects touching the image boundary were removed. The exact positioning of the segmentation outlines may vary between different segmentation approaches, and it is not straightforward to say if a 'tight' or a more 'loose' fit of an outline is more correct. This may introduce bias to the quantification

of precision and recall. We therefore adjusted the manually curated ground truth using simple dilations or erosions to make the area distribution of the cells in the ground truth as similar as possible to that of the output segmentation masks. In the label-based dilation, a square structuring element of size 3×3 was considered and the center pixel was assigned the label with maximum occurrence, excluding the background pixels.

After this adjustment of the ground truth image an object based segmentation evaluation was done. To evaluate the segmentation result per object, we created a list entry for every object. We thereafter considered each segment from the ground truth and found the segment with maximum overlap in the segmentation result from the corresponding algorithm and marked the corresponding entry in the list as visited. For every segment in the ground truth we find pixels that overlap with the associated segment from the respective algorithm and consider those pixels as true positive. Pixels in the ground truth segment that are not overlapping are considered as false negatives and similarly those segments not overlapping with the ground truth are considered as false positives. This was done for all the segments in the ground truth image. The ground truth segments that were not associated with any detected segment were considered as false negatives, and similarly, detected segments that failed to be associated with any of the segments from the ground truth image were considered as false positive cells. This ensured that we account for missing cells as well as false detections. We repeated this process on all the segmentation results and created a histogram of the performance metrics at 100 bin levels.

I. Cell Tracking

The segmented cells were tracked over time and cell lineages were created by a batch tracking algorithm that uses iterative application of the Viterbi algorithm [13]. The algorithm adds one cell at a time to the existing cell tracks in such a way that the probability of the tracks obtained is maximized. The algorithm uses the entire image sequence to create a track, thereby improving the tracking accuracy. In addition to tracking cells, the algorithm can handle events like cell division, cell death, cells entering and leaving the field of view, multiple cells present in a single segment, and cells that are missing in some frames. The basic parameters that were set for tracking are shown in Table II. The algorithm normally uses probabilities of cell migration events that are estimated by assuming that the centroid of a cell in one frame follows a Gaussian distribution centered on the centroid of the cell in the previous image. This does however not work well for rod shaped bacteria, as the centroids of adjacent bacteria can be very close to each other. To overcome this problem, we replaced the migration probability with the Jaccard similarity index of the binary masks of the segmented regions in the two images. The Jaccard similarity index is computed as the number of pixels in the intersection of the regions divided by the number of pixels in the union of the regions, and therefore takes values between 0 and 1.

TABLE II
PARAMETERS SET FOR TRACKING ALGORITHM

Parameter	Value
Probability that a segment contains 0 cells	0.000001
Probability that a segment contains 1 cell	0.93
Probability that a segment contains 2 cells	0
Probability of mitosis	0.20
Probability of cell death	0
Standard deviation of the cell velocity in pixels per frame	20

J. Cell Track Analysis

The tracking algorithm tracks cells over time from the first frame to the last frame. For our present experiments, we want to analyze the cell growth over time from the birth of a cell to its division. Due to noise and the complexity of the dataset, many tracks are generated, and not all of them are correct. Rather than keeping all tracks, we try to sort out correct tracks based on a priori information. Out of all the tracks that were generated, we filtered out only those tracks that start with a cell division and end with the next cell division. We used an area measure to confirm the cell division at the beginning and at the end of each track. At the beginning of the track we check if the area of the parent cell is within a tolerance level to that of the sum of the areas of the current cell and its sibling cell. Similarly, at the end of the track, we check if the area of the current cell at the end of the track is within a tolerance to the sum of the areas of the child cells. The error for parent and child cells are calculated as follows

$$child\ error = \left| 2 - \left\{ \frac{final\ area}{children\ area} + \frac{children\ area}{final\ area} \right\} \right|$$

$$parent\ error = \left| 2 - \left\{ \frac{final\ area}{siblings\ area} + \frac{siblings\ area}{final\ area} \right\} \right|$$

If both of these values were within a tolerance of 1% (found experimentally), we considered that track as a candidate track. Due to errors in the detection of cell divisions, and sudden movement of cells, some tracks, even though satisfying the above criteria, may not be representing the growth of a particular cell and the resulting tracks jumps from one cell to another. We therefore included an additional evaluation of the filtered tracks, as described below.

K. Evaluation of Track Quality

To analyze the growth of a cell, it is sufficient to analyze the increase in major axis length of the object, since the widths of all the cells are nearly constant. Cells grow more or less exponentially over time, meaning that the major axis length of a cell is approximately a straight line in logarithmic scale. To further evaluate the correctness of a track, we evaluated the noise level of this line using RANSAC [24] line fitting. Next we found the mean of the absolute difference of the estimated line (in logarithmic scale) and the observed major axis values. All the tracks were sorted in increasing order of this mean error value, and the tracks that had errors less than a specific threshold were considered for further post-tracking

segmentation correction as described below. This threshold was chosen based on the application and the acceptable error tolerance. In this experiment we set the value to 3 pixels.

L. Post-tracking Segmentation Correction

After the final quality control, some of the high-quality tracks may still contain single frame outliers where the segmentation algorithm has failed. We corrected for such segmentation errors using the assumption that the cells grow very little from one frame to the next frame. We considered a segment as erroneous if the absolute difference between the major axis length of the segment and the estimated major axis length by RANSAC fitting was above 10 pixels (found experimentally). To replace the erroneous segmentation result, we found the segment from the previous and next image. Then an average segment out of the two was calculated and placed in the image, as illustrated in Fig. 6. To find the average segment, we needed to find three things; 1) average position, 2) average orientation and 3) average shape. Average position was found as the mean position of the centroids of the two segments while average orientation was found by the average orientation of the major axes of the two segments. Then, to find the average shape, we first rotated both the segments such that their major axes were parallel to the x-axis and the centroids located at the middle of the image. Then signed distance transforms were performed on both these images. Finally, the average image was obtained as the average of the two signed distance transforms. This final averaged segment was placed in the erroneous frame replacing the erroneous segment.

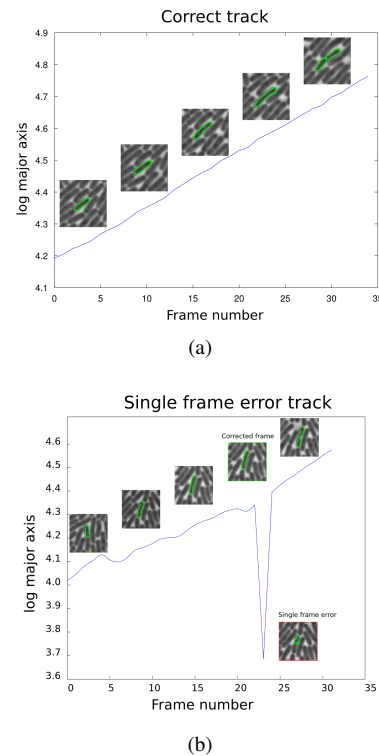


Fig. 6. Post-tracking segmentation correction: a) Track showing cell growth without any error and b) Track with single frame error and corresponding corrected frame.

III. RESULTS AND DISCUSSION

A. Segmentation Results

We compared the segmentation results of our Curvature Based Approach, referred to as CBA, with those of *MicrobeTracker* [5] and *MAMLE* [7]. As it has already been shown that *MAMLE* outperforms *Schnitzcells* [6] under many circumstances, we did not compare against this approach. We evaluated the segmentation performance on three *E. coli* phase contrast images: one from the *MicrobeTracker* example data, one from *Schnitzcells* (which was also used by the authors of *MAMLE*), and one from our own dataset. The *MicrobeTracker* and *Schnitzcells* examples are obtained for *E. coli* cells growing as largely isolated cells or micro-colonies on agarose pads whereas our dataset shows densely growing cells in PDMS microfluidics. It is important to note that neither *MicrobeTracker* nor *Schnitzcells* was developed to handle data of the type we have in our microfluidics experiments and that relative performance can vary depending on specific imaging conditions. We also evaluated the performance on one fluorescent image from our own dataset. Here, we inverted the fluorescence microscopy images and then processed them in the same way as the phase contrast images. We manually created ground truth for all the images. Selected regions of raw images and their corresponding ground truths are shown in the two top rows of Fig. 7. Our proposed method outputs the segmentation result as a binary image while the *MicrobeTracker* output is in the form of boundary coordinates. Therefore, special care was needed when performing the evaluation. The contours of two nearby cells detected by *MicrobeTracker* overlapped in some cases, and to avoid considering the overlapping regions as new segments we evaluated one contour at a time, converting each contour to a binary image by dilation followed by hole filling and erosion.

We also wanted to evaluate the improvement in performance of CBA with and without intensity filtering and contour smoothing, resulting in a total of five algorithms; 1) CBA, 2) CBA including intensity based filtering, 3) CBA including intensity filtering and smoothing, 4) *MicrobeTracker*, and 5) *MAMLE*. The results are as shown in Fig. 7.

Fig. 8 shows that the area distributions are maintained prior to evaluation of segmentation results as discussed in the methods section. Evaluation results are shown in Fig. 9. To summarize the comparison of the segmentation algorithms, we tested a range of thresholds on the F-score to quantify what proportion of the cells was correctly identified at different requirements on achieved F-score. We finally found the average F-score for each of the methods and the results are as shown in Table III. The results show that the performance of our method is comparable to that of the state-of-the-art methods.

The performance of CBA was only marginally improved when adding intensity based filtering and contour smoothing. However, the contour smoothing may be necessary for applications where exact edge positioning is crucial, e.g., when relating the position of sub-cellular signals to the cell edge.

Processing time was reduced by limiting the number of thresholds tested when searching for a locally optimal threshold. We have therefore evaluated the performance of our

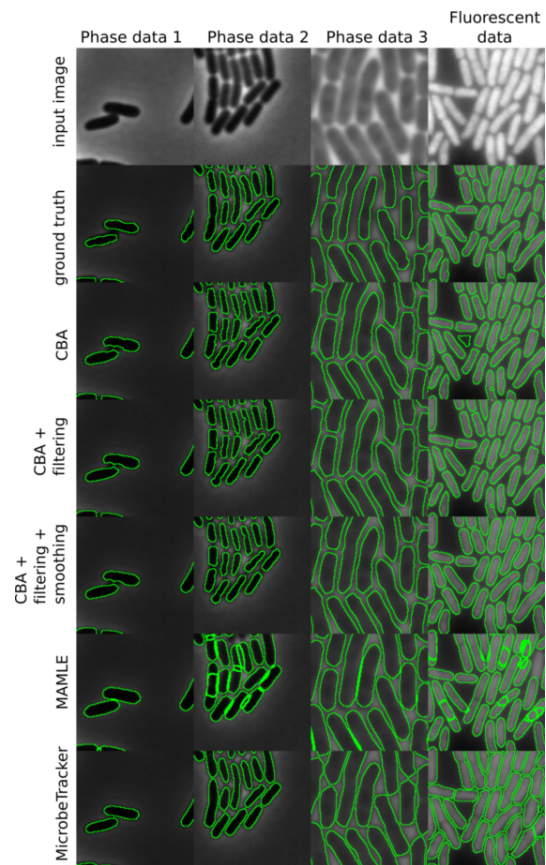


Fig. 7. Zoomed-in segmentation results: phase data 1 is from the *MicrobeTracker* dataset, phase data 2 from *Schnitzcells* dataset, phase data 3 and fluorescent data from our dataset. Phase data 1 and 2 are from experiments on agarose pads whereas phase data 3 and fluorescent data are from a growth medium in a PDMS device.

TABLE III
AVERAGE F-SCORE PERCENTAGE FOR DIFFERENT METHODS ON 4 DATASETS.

	Phase data 1	Phase data 2	Phase data 3	fluo data
CBA	87.49	78.64	85.77	85.56
CBA+Filt	86.95	78.64	87.56	85.82
CBA+Filt+Smooth	86.83	78.63	87.63	84.38
<i>MicrobeTracker</i>	79.49	71.29	62.39	55.03
<i>MAMLE</i>	84.90	82.00	81.95	81.71

algorithm in relation to the number of threshold levels tested. We calculated the average F-score while varying the number of tested threshold levels (around mean) from 2 to 20. The result is as shown in Fig. 10. The results shows that the performance is nearly stable around 10 for densely packed cells while parameter tuning is required for datasets with sparse cells. In Fig. 10, the average F-score for phase data 1 dropped because of false positive detections when the window size was too large.

We also evaluated the stability of the threshold for max projected images by varying the threshold from 0.5 to 1.0 and found average F-scores for segmented objects for the 4 datasets. The result shows that the average F-score is stable

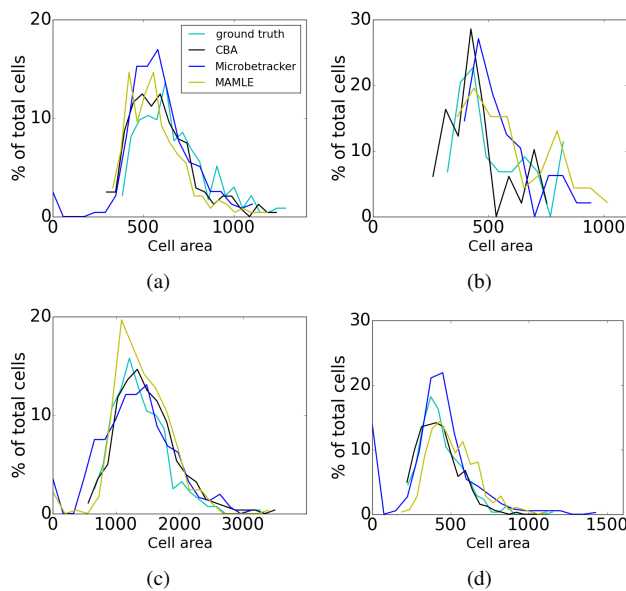


Fig. 8. Area distribution of segmented cells for a) phase data 1, b) phase data 2 c) phase data 3 and d) fluorescent data.

around the region of the threshold at 0.75 and is as shown in Fig. 11.

Though our method did well on 4 datasets there were some instances where our method failed to find the cells properly. This occurred due to the parameter limits that we use in our study. It is possible that some irregularly shaped cells or unusually long cells may be missed in the detection. Two such examples are shown in Fig. 12

B. Run Time Evaluation

Our algorithm, CBA, was implemented in Python as a multithreaded program. We compared the execution speed of the methods on the four datasets. The time taken by each method on the four images in the dataset is shown in Fig. 13. The experimental results show that the execution speed depends on the number of cells present in the image. It was found that our method, CBA, is an order of magnitude faster than the state-of-the-art methods. The results show that CBA is 10 times faster than *MicroTracker* and 8 times faster than *MAMLE* on average. We performed the speed evaluation on a laptop with a dual core, 2.7GHz, Core i7 processor and 16GB RAM running on Windows 7.

C. Tracking Results

To further evaluate the usability of the segmentation results produced by our proposed method, we tested its performance on a large time-lapse experiment. We applied our segmentation algorithm to time-lapse images from a dataset comprising 500 images with approximately 250 cells per image. 500 frames correspond to 250 min in real time. Cell tracking was done using the Viterbi algorithm as described in the Methods section, and we obtained a total of 10259 tracks. The tracks were then subjected to filtering to extract candidate tracks. The track analysis stage found 1529 tracks satisfying the criteria

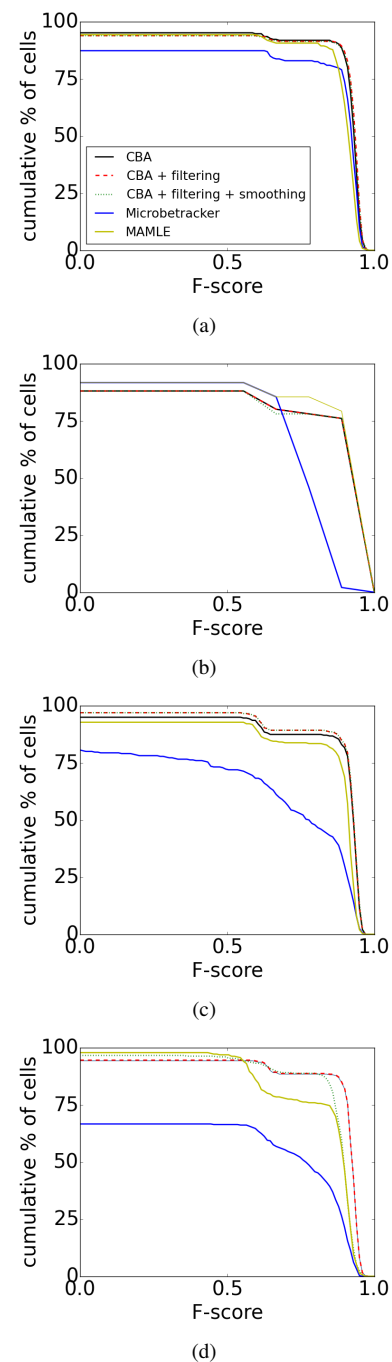


Fig. 9. Cumulative % of cells above the particular F-score for a) phase data 1, b) phase data 2, c) phase data 3 and d) fluorescent data.

of having cell division at the beginning and also at the end. Since we consider cell division at the start and end of the track, essentially all tracks including the first or last 30 min were discarded. All the 1529 tracks were sorted in increasing order based on RANSAC fit error, as shown by the dashed line in Fig. 14 (note the logarithmic scale on the fit error). We further searched for and corrected segmentation errors in the tracks. The solid line shows the improved result, where a total of 1238 tracks fell below the 3 pixel error threshold prior to track correction, and 1274 fell below the threshold afterwards. In frames 100 to 400, 40% of the identified cells are assigned

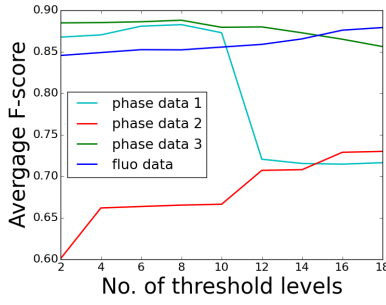


Fig. 10. Plot showing the stability of number of threshold levels required for 4 datasets, which is nearly constant around 10 for datasets with dense cells.

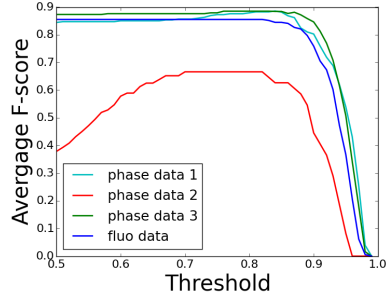


Fig. 11. Plot showing the stability of threshold on max projected segmented stack for 4 datasets, which is nearly constant around the region of 0.75.

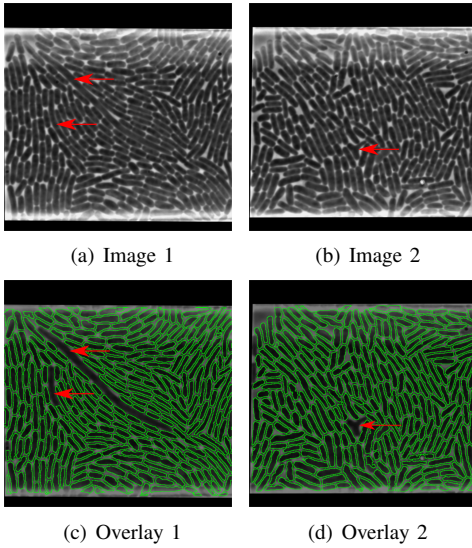


Fig. 12. Missing cells due to limits of parameter setting: a) input image with unusually longer cells with c) the segmentation result and b) shows an *E. coli* cell with irregular shape and the corresponding segmentation result in d). Red arrows show the missing cells.

to a track.

D. Comparison of Growth Rate at Different Treatments

The second experiment included in this work was aimed at testing if the proposed method could be used to find differences in the growth rate of *E. coli* cells in different media. The first dataset was obtained from *E. coli* cells using glycerol as medium and the second dataset was obtained using glucose as medium. We wanted to know how fast the cells

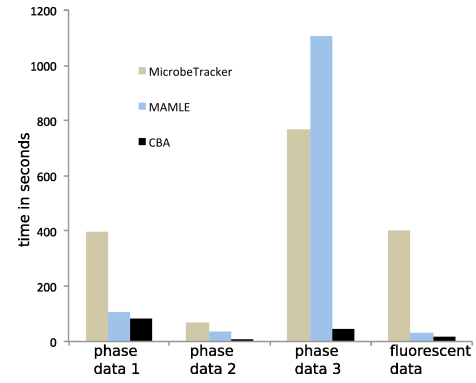


Fig. 13. Execution times of *MicrobeTracker*, *MAMLE* and *CBA* on four datasets.

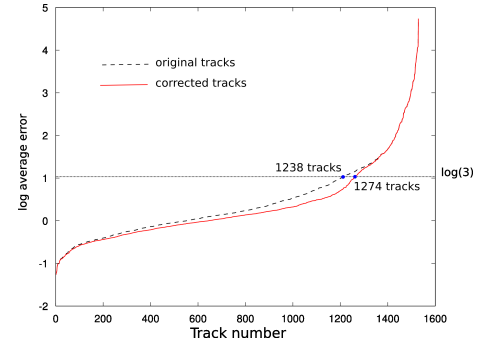


Fig. 14. RANSAC fit error of the major axis length in cell tracks, for the original segmentation and the error corrected segmentation, shown in logarithmic scale.

grow and divide in these two media. We found that the growth rate measured as the slope of the RANSAC fitted line was $1.2 \times 10^{-2} \pm 2.1 \times 10^{-3}$ pixels/frame in logarithmic scale (mean \pm standard deviation) for cells in glucose (red) and $9.2 \times 10^{-3} \pm 1.5 \times 10^{-3}$ pixels/frame in logarithmic scale for cells in glycerol (blue), showing that cells in glucose grow faster than those in glycerol. There are some cells that grow faster in glycerol and cells that grow slower in glucose as can be seen in Fig. 15.

IV. CONCLUSIONS AND FUTURE WORK

In this work, we developed a fast and robust *E. coli* cell segmentation method that can be used as input for tracking cells in time-lapse microscopy images. We also showed that our method works better than the state-of-the-art methods in terms of speed with comparable accuracy. This is particularly important when analysing data from the increasingly popular microfluidic experiments where supply of nutrients and crowding by cells no longer limits the extent of the experiments. Faster and accurate algorithms are needed to analyse the data on a similar time scale as the acquisition time. Given the possibility to process much larger data sets and deeper cell lineage trees, we can start addressing a new level of biologically relevant questions such as how gene expression in one generation influences the fitness of the offspring several generations later or how cell to cell variation

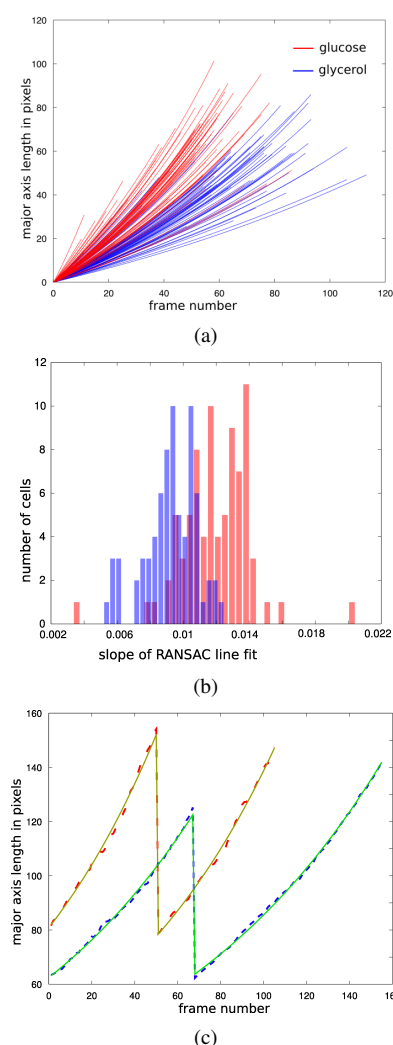


Fig. 15. Comparison of growth rates: a) Change in major axis length over time for two datasets with cells in glucose (red) and glycerol (blue). b) Histogram of slope found by RANSAC line fit for two datasets. c) A sample lineage of cells grown in glucose (raw data in red and RANSAC in yellow) and glycerol (raw data in blue and RANSAC in green).

in growth rate depends on availability of nutrients, antibiotics and cell to cell interactions. Since small differences in growth rate can lead to large differences in biological fitness due to exponential amplification over generations, it is particularly important to have robust and automatic ways to segment cells and determine growth rates, such that the results from two different experiments not mainly depend on parameter tuning. Here we have shown that it is possible to obtain such results.

In the future, we would like to speed up the presented algorithm further and include sub-pixel precision in segmentation results with possible GUI option.

ACKNOWLEDGMENT

The research leading to these results has received funding from the European Research Council under the European Union's Seventh Framework Programme (FP/2007-2013) / ERC Grant Agreement n. 616047 to Johan Elf, the Swedish research council, grant 2012-4968, and Swedish strategic research program eSSSENCE, to Carolina Wählby.

REFERENCES

- [1] M. B. Elowitz, A. J. Levine, E. D. Siggia, and P. S. Swain, "Stochastic gene expression in a single cell," *Science*, vol. 297, pp. 1183–1186, 2002.
- [2] Y. Taniguchi, P. J. Choi, G. Li, H. Chen, M. Babu, J. Hearn, A. Emili, and X. S. Xie, "Quantifying e. coli proteome and transcriptome with single-molecule sensitivity in single cells," *Science*, vol. 329, pp. 533–538, 2010.
- [3] Q. Wu, F. Merchant, and K. Castleman, *Microscope Image Processing*. Massachusetts: Academic Press, 2008.
- [4] M. Wallden and J. Elf, "Studying transcriptional interaction in single cells at sufficient resolution," *Current Opinion in Biotechnology*, vol. 22, pp. 81–86, 2011.
- [5] O. Sliusarenko, J. Heinritz, T. Emonet, and C. Jacobs-Wagner, "High-throughput, subpixel precision analysis of bacterial morphogenesis and intracellular spatio-temporal dynamics," *Mol. Microbiol.*, vol. 80, pp. 612–627, 2011.
- [6] J. W. Young, J. C. W. Locke, N. R. A. Altinok, T. Bacarian, P. S. Swain, E. Mjolsness, and M. B. Elowitz, "Measuring single-cell gene expression dynamics in bacteria using fluorescence time-lapse microscopy," *Nat. Protocols*, vol. 1, pp. 80–88, 2012.
- [7] S. Chowdhury, M. Kandhavelu, O. Yli-Harja, and A. S. Ribeiro, "Cell segmentation by multi-resolution analysis and maximum likelihood estimation (MAMLE)," *BMC bioinformatics*, vol. 14, p. S8, 2013.
- [8] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Sys. Man. Cyber.*, vol. 9, pp. 62–66, 1979.
- [9] C. Zimmer, E. Labruyere, V. Meas-Yedid, N. Guillen, and J.-C. Olivo-Marin, "Segmentation and tracking of migrating cells in videomicroscopy with parametric active contours: a tool for cell-based drug testing," *IEEE Trans. Medical Imaging*, vol. 21, p. 12121221, 2002.
- [10] O. Dzyubachyk, W. A. van Cappellen, J. Essers, W. J. Niessen, and E. Meijering, "Advanced level-set-based cell tracking in time-lapse fluorescence microscopy," *IEEE Trans. Medical Imaging*, vol. 29, pp. 852–867, 2010.
- [11] R. Bise, Z. Yin, and T. Kanade, "Reliable cell tracking by global data association," in *Proceedings of IEEE International Symposium on Biomedical Imaging: From Nano to Macro*. IEEE, 2011, pp. 1004–1010.
- [12] F. Jug, T. Pietzsch, D. Kainmüller, J. Funke, M. Kaiser, E. van Nimwegen, C. Rother, and G. Myers, "Optimal joint segmentation and tracking of Escherichia coli in the mother machine," in *Proceedings of Bayesian and graphical Models for Biomedical Imaging*, ser. Lecture Notes in Computer Science, M. J. Cardoso, I. Simpson, T. Arbel, D. Precup, and A. Ribbens, Eds., vol. 8677. Berlin: Springer, 2014, pp. 25–36.
- [13] K. Magnusson, J. Jaldén, P. Gilbert, and H. Blau, "Global linking of cell tracks using the Viterbi algorithm," *IEEE Trans. Med. Imag.*, vol. 34, no. 4, pp. 1–19, 2014.
- [14] P. J. Choi, L. Cai, K. Frieda, and X. S. Xie, "A stochastic single-molecule event triggers phenotype switching of a bacterial cell," *Science*, vol. 322, pp. 442–446, 2008.
- [15] G. Ullman, M. Wallden, E. G. Marklund, A. Mahmutovic, I. Razinkov, and J. Elf, "High-throughput gene expression analysis at the level of single proteins using a microfluidic turbidostat and automated cell tracking," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 368, pp. 1–8, 2012.
- [16] P. Hammar, P. Leroy, A. Mahmutovic, E. Marklund, O. G. Berg, and J. Elf, "The lac repressor displays facilitated diffusion in living cells," *Science*, vol. 336, pp. 1595–1598, 2012.
- [17] T. J. Willmore, *An introduction to differential geometry*. Oxford: Oxford University Press, 1959.
- [18] J. A. Thorpe, *Elementary topics in differential geometry*. New York: Springer-Verlag, 1979.
- [19] C. Woodford and C. Philips, *Numerical methods with worked examples: Matlab edition*. New York: Springer-Verlag, 2012.
- [20] W. Burger and M. J. Burge, *Principles of digital image processing: Core algorithms*. New York: Springer-Verlag, 2009.
- [21] E. Jones, E. Oliphant, and E. A. P. Peterson. (2001) Scipy: Open source scientific tools for python. [Online]. Available: <http://www.scipy.org/>
- [22] W. Eric. (2014) Full width at half maximum. [Online]. Available: <http://mathworld.wolfram.com/FullWidthatHalfMaximum.html>
- [23] P. Marquez-Neila, L. Baumela, and L. Alvarez, "A morphological approach to curvature-based evolution of curves and surfaces," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, pp. 2–17, 2014.
- [24] M. A. Fischler and R. C. Bolles, "Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography," *Comm. of the ACM*, vol. 24, pp. 381–395, 1981.



Sajith Kecheril Sadanandan born in 1983, received an M.Tech. degree in computer vision and image processing from Amrita Viswavidyapeetham University, India, in 2012. He is currently pursuing Ph.D. at Uppsala University, Sweden, in Carolina Wahlby's group. He worked at Center for development of advanced computing, India, from 2006 to 2010, developing image processing algorithms for automated cervical cancer screening. He also worked as an Assistant Professor at Department of Electronics and Communication, Amrita Viswavidyapeetham

University. His research interests include microscopic image processing and pattern recognition.



Özden Baltekin born in 1987, received an M.Sc. from University of Montpellier in 2012. He is currently a PhD candidate at Uppsala University, Department of Cell and Molecular Biology in Johan Elf's research group. Özden is doing research on the fitness consequences of bacterial cell-to-cell variation and on rapid detection of antibiotic resistance using microfluidic systems.



Klas E. G. Magnusson received an M.Sc. in Engineering Physics from KTH in 2011. In 2008 and 2009, he studied as an exchange student, and performed his masters thesis work at Stanford University, CA, USA. Klas is currently a Ph.D. student in the Department of Signal Processing at KTH. He works on segmentation and tracking of cells and subcellular particles in 2D and 3D time lapse microscopy. His main focus is on algorithms for track linking. At the International Symposium on Biomedical Imaging (ISBI) 2012, Klas received a

best paper award for his paper A batch algorithm using iterative application of the Viterbi algorithm to track cells and construct cell lineages. Klas participated in the ISBI Particle Tracking Challenge 2012 and in the ISBI Cell Tracking Challenges 2013, 2014, and 2015. He achieved outstanding results in the Cell Tracking Challenges, and for that he received Bitplane ISBI attendance awards in 2014 and 2015.



Alexis Boucharin born in 1987, received an engineering degree (M. Sc.) from CPE Lyon School of Engineering in 2012. During his studies he worked in academia developing algorithms for neuroscience and radiotherapy. After graduating he worked at the Aging Research Center in Stockholm developing tools related to brain MRI analysis. In October 2013 he arrived at Uppsala University to join the Elf lab research group led by Johan Elf. His main duties are developing tools and implementing algorithms to process and analyze images for microscopy ap-

plications.



Petter Ranefall born in 1968, received an M.Sc. from Uppsala University in 1992. In 1998 he defended his PhD thesis Towards Automatic Quantification of Immunohistochemistry Using Color Image Analysis at the Centre for Image Analysis, Uppsala University. After that he started working in industry developing image analysis software targeted for microscopy. He has also worked with image analysis, image processing and software development in relation to MRI and radiotherapy. In August 2013 he returned to Uppsala University to join the

Quantitative Microscopy research group led by Carolina Wahlby. His main focus is research support on image analysis for microscopy applications.



Joakim Jaldén (S'03-M'08-S'13) received the M.Sc. and Ph.D. in electrical engineering from the KTH Royal Institute of Technology, Stockholm, Sweden in 2002 and 2007 respectively. He studied at Stanford University, CA, USA, from September 2000 to May 2002. From July 2007 to June 2009 he held a post-doctoral research position at the Vienna University of Technology, Vienna, Austria. He is currently a tenured Associate Professor at KTH, serves as an associate editor for the IEEE Transactions in Signal Processing, and is a member

of the IEEE SPCOM Technical Committee.

Joakim was awarded the IEEE Signal Processing Society's 2006 Young Author Best Paper Award. He also received the Ingvar Carlsson Career Award in 2009 from the Swedish Foundation for Strategic Research. Recent work includes signal processing for biomedical data analysis, and automated tracking of (biological) cells. Early work in this field was awarded a conference best paper award at IEEE ISBI in 2012.



Johan Elf born in 1975, is Professor in Biological Physics at Uppsala University. Together with his interdisciplinary research team, Johan is pioneering single-molecule tracking methods by means of fluorescence microscopy in living cells. After obtaining his PhD on a theoretical subject in 2004, Johan Elf spent a more experimentally oriented postdoc period in the lab of Sunny Xie at Harvard University, developing the first methods for studying transcription factor binding at the level of single molecules in living cells. In the years following his return to

Uppsala in 2007, Johan has focused his efforts on the study of gene regulation in bacteria, successfully combining theoretical and computational biology with experimental work and methods development to tackle challenging biological problems related to the central dogma of molecular biology.



Carolina Wahlby born in 1974, received a MSc in molecular biotechnology and a PhD in digital image analysis from Uppsala University, Sweden, in 2003. She carried out PostDoc research at the Dept. of Genetics and Pathology and joined the Imaging Platform of the Broad Institute of Harvard and MIT in 2009, focusing on development of algorithms for high-throughput analysis of model organisms, such as *C. elegans* and zebrafish, contributing to the CellProfiler project. She became full Professor in Quantitative Microscopy at the Dept. of Information

Technology and SciLifeLab, Uppsala University, in 2014, was elected ISAC scholar by the Int. Society for Advancement of Cytometry 2014-2018, received the President's innovation award from the Society of Biomolecular Imaging and Informatics in 2014, and was awarded the Thérus prize by The Royal Society of Sciences at Uppsala in 2015 for pioneering the development of new computerized image analysis methods for use in life science research.