OCCLUDING CONVEX IMAGE SEGMENTATION FOR *E.COLI* MICROSCOPY IMAGES

Zoltán Kutalik[1,2]*, Moe Razaz*[1] *and József Baranyi*[2]

[1]Royal Society Wolfson Bioinformatics Research Laboratory, School of Computing Sciences, University of East Anglia, Norwich, UK, email: mr@cmp.uea.ac.uk

[2]Department of Food Safety Science, Institute of Food Research, Norwich, UK

ABSTRACT

State-of-the-art flow-chamber technology enables us to closely monitor individual growth of thousands of bacterial cells simultaneously and across time. These experiments provide us with spatio-temporal greyscale images from the early stage of growth. Due to a large number of cells and time points involved automated image analysis covering noise removal, cell recognition and occluding image segmentation becomes essential. In this paper we focus on occluding image segmentation. A novel convex hull based method has been devised by the authors, which is compared with previously published algorithms through testing on real and simulated images. Results clearly show that our convex hull based segmentation algorithm works better than the ones based on curvature.

1. INTRODUCTION

An innovative flow-chamber system was recently developed in [3]. Cells of *E. coli* were anchored to the surface of a flow chamber set at the desired environmental conditions. They were observed with a dark-field microscope equipped with a 4x objective. A CCD camera was mounted on to the microscope and the digital signal was transferred to a PCI interface card. An image analysis program controlled the camera to grab an image every 5 minutes. This technique is a significant innovation showing several features of the transition process of individual cells, from the lag to exponential phase that have not been observed before.

Flow chamber experiments were set up and carried out at the Institute of Food Research, Norwich, UK.

We developed a number of algorithms for processing flow chamber images. First we applied image stretching and 'power-law' transformation [4]. Noise removal was tailored to the nature of the images. In order to reduce falsely interpreted cells, we applied a Vincent Dome Transformation to filter background noise [8]. The advantage of this is being insensitive to small background noise and incorporating spatial information into the filtering. To determine the cells, region growing algorithm was used combined with 'Laplacian of a Gaussian' filter [4]. The size of the filter was chosen to be 5, and the standard deviation 2.

Since the Laplacian filter introduces unwanted sporadic noise in the background, median filtering is used to overcome this problem. Subsequently the cores of the cells (so called *seeds*) are defined and iteratively grown. Thickening is controlled in two ways: a neighbouring pixel is annexed to the core if (1) its intensity is higher than a specific threshold and (2) its median-filtered Laplacian is zero. The threshold was chosen in such a away that the total area of the thickened image should not exceed more than 1.5 times that of the seeds. During post-processing we excluded cells (1) touching the margin of the screen, or (2) being small, and selected the ones being too concave.

This algorithm completely utilizes our prior knowledge about the images. Namely, we know (1) what belongs to a cell, (2) what belongs to the background; (3) smoothed Laplacian edge detection combined with (4) intensity and (5) spatial information aids us to decide which pixels can be annexed to the seeds.

Since thousands of cells are inoculated on a slide, cell occlusion is inevitable. This problem can affect up to 10 % of the cells, therefore we devised a new algorithm based on convex hull to separate them. Our proposed approach outperformed traditional curvature based methods when applied to *E.coli* images. Moreover, extensive simulation analysis pointed out that our novel idea is far more effective in cases of (1) acute angle occlusion, (2) multiple object occlusion, and (3) overlapping objects with noisy boundary.

The outline of the paper is as follows. In section 2 we provide a brief literature review on relevant occluding image segmentation and describe our novel method. Next we present our results in Section 3 and finally summarize our conclusions in Section 4.

2. OCCLUDING IMAGE SEPARATION

Overlapping or touching cells are regarded as one cell by automatic image processing leading to erroneous results. Therefore it is essential to devise automatic cell separation algorithm to identify cell occlusion. An example of *E.coli* cell occlusion and its automatic segmentation is illustrated in Figure 1.

2.1 Previous Work on Occluding Image Segmentation

Most of the work so far in the literature on occluding image segmentation has been motivated by overlapping chromosome segmentation (e.g. [7, 5, 1, 10, 6]). Since the shape

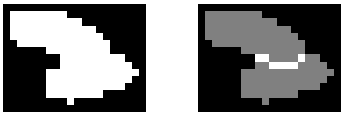

Figure 1: Original and segmented binary image of two occluding *E.coli* cells.

The authors would like to thank Carmen Pin for her collaboration.

of chromosomes differ largely from that of *E.coli* cells, most of those algorithms are not suitable for our case. However the first part of chromosome segmentation methods is to find saliency/concavity points of interest, which is common with our purpose. Charters *et al.* [1] used object skeleton to manually approximate Ji's method [5]. In the case of small objects (our *E.coli* cells are typically less than 100 pixels in area) with noisy boundary the skeleton proved to be useless due to the large number of branches caused by the noise. Vast majority of other methods use smoothed curvature, *k-curvature*, as a measure of concavity (e.g. [10, 6]). Pale path cutting method [7] cannot be applied for our images since strong cell overlaps were present even at the highest intensity levels. The most relevant method to our cell separation problem was that presented in [9], where segmentation of overlapping ellipsoid shaped objects (grain kernels) was addressed. Visen *et al.* [9] computed smoothed curvature of window size 7, and then applied a 25% frequency width, Fourier domain, low-pass filter to smooth the resulting curvatures even further. Finally, close points with large negative curvature are connected to form a cutting line. The most advanced approach was devised by Ji [5] who combined skeleton, curvature and convex hull to obtain chromosome overlaps.

Our new method starts similarly as Ji's algorithm, but exploits more the convex hull information, rather than curvature and skeleton, since those quantities are prone to be erroneous in case of small and noisy images.

2.2 The new algorithm

Touching cells can be detected by convexity analysis. The best measure for this – according to our experience – is the maximum of the distances between the *image boundary points* (*bI*) and the closest *boundary point of the convex hull* (*bcI*). We have

$$\text{Convexity measure of image I} = \max\{d(z,bcI)|z \in bI\} \quad (1)$$

where $d(z_1,z_2)$ is the usual Euclidean distance of two objects. In the following we consider a rectangular segment of the original image containing only one connected object with convexity measure greater than 2. Cell occlusion is present even if the images are thresholded at the highest possible intensity level. Therefore the original greyscale image in itself is unsuitable for segmentation purposes. For this reason we propose a transformation of the thresholded binary image into a shape-based greyscale image. In the transformed image the intensity of an object pixel is the difference between (1) its distance from the boundary of the convex hull of the object (cell) and (2) the distance from the boundary of the object. The closer a point is to a concavity point on the boundary the larger its intensity will be in the transformed image. For details see Figure 2. Line across P and C is the rupture, two curves intersecting in C are fragments of the boundary of the image, while the line through T_1 and T_3 is the boundary of the convex hull. The transformed intensity value for point B_2 is determined as follows. Its distance from the boundary of the convex hull is $\overline{B_2T_3}$, that from the boundary of the object is $\overline{B_2C}$. Hence the transformed intensity of B_2 will be $|\overline{B_2T_3}| - |\overline{B_2C}|$. Moreover it is obvious that if a point P is on the rupture line, then any other point in the perpendicular direction (e.g. B_1, B_2) will have lower intensity value than P.

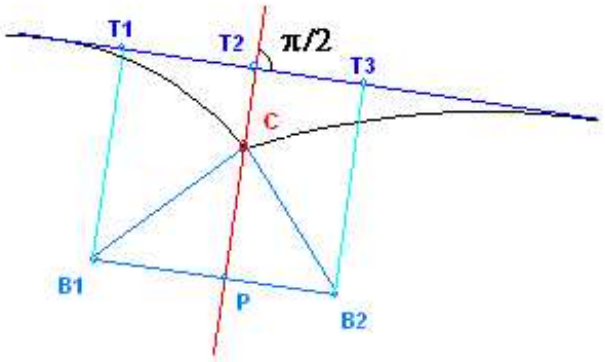

Figure 2: Geometric explanation of image transformation.

The algorithm starts at a boundary point (x,y) which will be the first element of a possible rupture path, then it selects an arbitrary neighbour belonging to the object, say a_1 (Figure 3). If a_1 is greater than its two neighbouring elements in normal direction to the $\overrightarrow{(x,y)a_1}$ vector, then a_1 is added to the rupture. In case there are more than one neighbour of (x,y) that satisfy the criterion the algorithm chooses the one with greatest intensity. Say a_1 is the chosen extension of the rupture, the allowed direction will be $\overrightarrow{(x,y)a_1}$. In the next step only those neighbours of a_1 may be considered which are within the $\pm 45°$ range of the allowed direction (marked with dashed arrows in Figure 3). The steps are repeated until the path reaches the boundary again.

The pseudo-code of our algorithm is available from the authors on request, but here to save space we rather illustrate the idea through a simple example (Fig. 4). First image transformation is applied to the binary image (Fig. 1) according to Figure 2. Having obtained the intensity image we iteratively expand the rupture which starts off from a boundary point. Details are found in Figure 4. The binary image is transformed into intensity image shown in the figure, "-1" represents background pixel. Then starting off from a boundary point we are looking for ridge-like path across the image. One can see as the rupture evolves. The new searching direction can decline by a maximum 45° from the previously selected direction. For instance, starting off with '4' we compare it with '3.97' and '3.47'. Since it is greater than both we move to the perpendicular direction ± 45°. If there are more than one option to move on we choose the one with

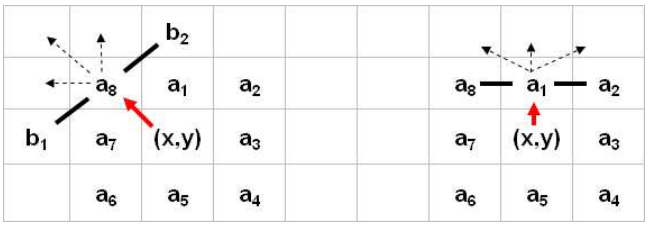

Figure 3: Solid arrow indicates an allowed direction and lines lead to points involved in comparison. At the right hand-side table, if $I(a_1) \geq \max\{I(a_8),I(a_2)\}$ then pixel a_1 will be annexed to the rupture line. Dashed arrows indicate the allowed directions in the next step.

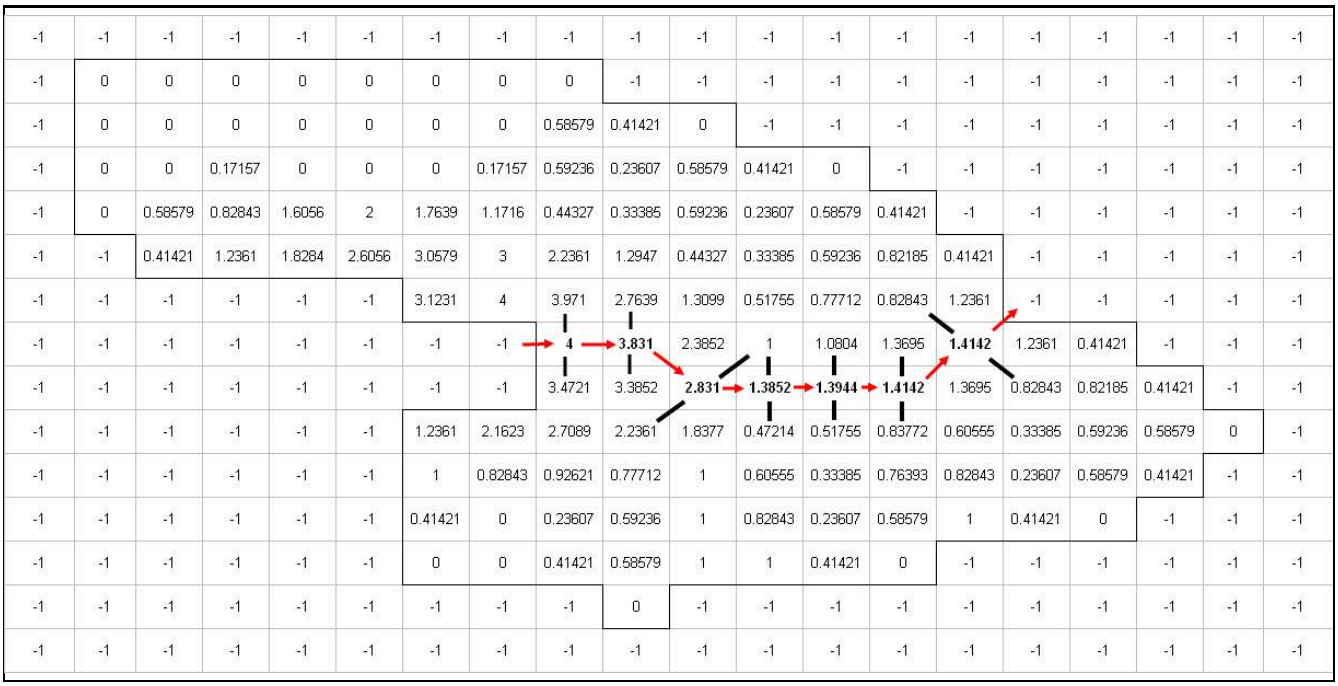

Figure 4: Illustration of our new algorithm in a real *E.coli* cell occlusion example. Intensity image is obtained by our transformation applied to Figure 1.

largest intensity. So we pick '3.83' since it is bigger than both '2.76' and'3.38'. In the next step we only find peak in the -45° direction: '2.83' (since it is greater than both '2.23' and '1'). The algorithm proceeds until the rupture reaches another boundary point. This will define a complete cut.

The algorithm is run for all possible boundary points. Every complete rupture path has a score defined as the product of (1) average intensity of the pixels of the rupture and (2) the minimal solidity (area / convex area) of the newly arising objects (after cutting up the original object along the rupture path). In case we end up with having more than one complete rupture path in an image, we select the one with the highest score. Finally, the algorithm is repeated for all the newly arising objects. When the extent of overlap is high (X-shaped overlaps), our algorithm over-segments the object, thus some segmented parts may need to be rejoined. Since the shape of *E.coli* cells is approximately ellipsoid, cohesive parts can be determined by Hugh-transform applied to detect elliptical arcs [2].

3. RESULTS

We compared our algorithm with curvature based methods [9, 6, 10] through both real image data and simulated examples.

In the case of the *E.coli* data set there were 80 objects where cell occlusion was detected automatically, out of which 14 were unrecognizable even by eye due to the high noise level. Out of the remaining 66 cases 41 (62.1%) could be detected by curvature based methods. Curvature computation was applied with different window sizes (3, 5, 7) and for each cell occlusion we picked the best performing window size. Our rupture finding algorithm identified 59 (89.4%) correct separations.

Further extensive testing was carried out using simulation. Two or more randomly placed occluding ellipses were superimposed. We tested both algorithms for different degrees and positions of occlusion and noise. We demonstrate the advantage of our idea thorough three different aspects as described in (i), (ii) and (iii) below.

(i) The rupture can cut through the object if it is well-justified from at least one end of the path, even if the other end leads to a convex boundary section. When the overlap is near the end of the major axis of both ellipses which incline acute angle, curvature based methods fail to work since they can find only one cut point on the boundary. This kind of overlap is illustrated in Figure 5. Curvature based methods are unable to detect the proper cut in case of such overlaps, while our new method finds the correct rupture as shown in the figure. To verify our theory we simulated 120 occluding ellipses with close endpoints and small inclined angles. On 14 instances there were complete overlaps, so we considered the remaining 106 images for comparison. The best working window size curvature computing method [9] managed to separate correctly 42 out of the 106 occlusions (39.6%), our new method on the other hand found the right cut for 72 images (67.9%).

(ii) When more than 2 objects are overlapping, curvature based methods fail to find the correct pairs of points (with

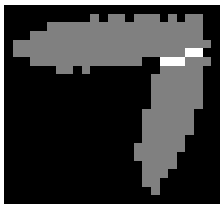

Figure 5: An example of type (i) occlusion undetectable by curvature based methods.

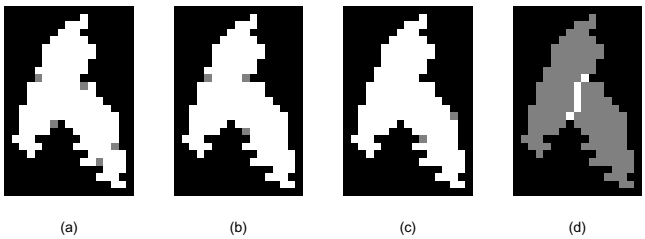

Figure 6: Figure (a)-(c) illustrates the effect of window size (3, 9, 17) on the curvature method results for noisy images. Our novel method (d) however, is much less affected by noise.

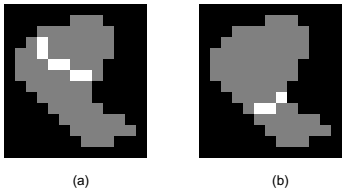

Figure 7: Two most probable rupture cuts for substantial *E.coli* overlap.

high curvature) to form cuts. The performance of our new algorithm, on the other hand is not affected by multiple overlaps since the intensity peaks on the transformed image are isolated, thus ruptures will be independent of each other.

(iii) Finally, we investigated the effect of noisiness of the object boundary on performance. The problem is illustrated in Figure 6. In these images (a-c) grey pixels indicate suggested convexity points by the corresponding curvature algorithm. It is a compromise between finding unambiguous cut by increased smoothing (lowpass filtering) window, and determining the possible cut point with most accuracy by decreasing the window size. The figure exemplifies that for small window size the curvature method [9] offers too many possible cut points, while for a larger window size its solution becomes inaccurate. 120 randomly simulated images were generated with extremely noisy boundary (the noise was allowed to be one fourth of the object size). The image size was uniformly 40-by-40 pixels. 10 of these images produced complete overlap, thus those were excluded from further analysis. Due to the large noise ratio an increased window size (15) was used for more reliable curvature determination. A cut was called accurate if both endpoints fell within 3-pixel-radius of the ideal cut point (which was virtually determined). Surprisingly only 14 images (12.7%) were segmented properly within the accuracy limit by the curvature based method. Our proposed method was able to accurately segregate 92 out of the total 110 images (83.6%).

A further advantage of ruptures appears when two objects are overlapping substantially. Different ruptures with (almost) the same starting (boundary) point suggest how the two objects would have continued if they had been separate. This is illustrated by a real image in Figure 7. In Figure 7(a) (whose cut is suggested by our method) the bottom cell is shown completely. While in the case of the other rupture path [Figure 7(b)] (second most probable by our method) starting from nearly the same point the upper cell can be seen completely. This is verified in the image taken at the next time point, where the cells drifted away from each other.

Our proposed method needs thresholding at no point, therefore works fully automatically regardless of the image type and the magnitude of noise.

The drawback of our proposed rupture finding algorithm is its computational complexity. For an n-by-n sized image our method needs $O(n^3)$ steps to run, while curvature based methods perform only $O(n)$ computations. Therefore for large images with small noise, where both algorithms perform equally well traditional curvature based methods are preferred, whereas for relatively small images with jagged boundary our method is recommended.

4. CONCLUSIONS

Our proposed convex hull based method outperforms the existing curvature based algorithms in the literature when applied for occluding convex image segmentation. The main advantage of our method manifests itself significantly in cases of (1) acute angle overlapping, (2) multiple image overlapping and (3) in presence of high noise.

REFERENCES

[1] G. Charters and J. Graham "Disentangling chromosome overlaps by combining trainable shape models with classification evidence." *IEEE Transactions on Signal Processing*, vol. 50, pp.2080–2086, 2002.

[2] C. Daul, P. Graebling, and E. Hirsch "From the hough transform to a new approach for detection and approximation of elliptical arcs." *Computer Vision and Image Understanding*, vol. 72, pp. 215–236, 1998.

[3] A. Elfwing, Y. Le Marc, J. Baranyi, and A. Ballagi "Observing the growth and division of large number of individual bacteria using image analysis." *Applied Environmental Mircobiology* (In press), 2004.

[4] R. Gonzalez and R. Woods *Digital Image Processing*. New Jersey 07458: Prentice-Hall, 2nd edition, 2002.

[5] L. Ji "Decomposition of overlapping chromosomes." in *Automation of cytogenetics*, C. Lundsteen and J. Piper, eds. New York: Springer-Verlag, 1989, pp. 177–190.

[6] B. Lerner, H. Guterman, and I. Dinstein "A classification-driven partially occluded object segmentation (cpoos) method with application to chromosome analysis." *IEEE Transactions on Signal Processing*, vol. 46, pp.2841–2847, 1998.

[7] M. Popescu, P. Gader, J. Keller, C. Klein, J. Stanley, and C. Caldwell "Automatic karyotyping of metaphase cells with overlapping chromosomes." *Computers in Biology and Medicine*, vol.29, pp. 61–82, 1999.

[8] L. Vincent "Morphological greyscale reconstruction in image analysis: Applications and efficient algorithms." *IEEE Transactions on Image Processing*, vol 2., pp. 176–201, 1993.

[9] N. Visen, N. Shashidhar, J. Paliwal, and D. Jayas "Identification and segmentation of occluding groups of grain kernels in a grain sample image." *Journal of Agricultural Engineering Research*, vol. 79, pp. 159–166, 2001.

[10] A. Vossepoel "Separation of touching chromosomes." in *Automation of cytogenetics*, C. Lundsteen and J. Piper, eds. New York: Springer-Verlag, 1989, pp. 205–216.