City Research Online

City, University of London Institutional Repository

This is the accepted version of the paper.

This version of the publication may differ from the final published version.

SHAPE ANALYSIS AND TRACKING OF MIGRATING MACROPHAGES

José Alonso Solís-Lemus⋆ *Brian Stramer*† *Greg Slabaugh*⋆ *Constantino Carlos Reyes-Aldasoro*⋆

⋆School of Mathematics, Computer Science and Engineering, City, University of London, UK
†Randall Division of Cell & Molecular Biophysics, King's College London, UK

ABSTRACT

This paper describes an algorithm to iteratively segment, track and analyse the shape of macrophages from fluorescent microscopy image sequences. This process allows observation of shape variations as the cells migrate. The algorithm identifies and separates overlapping and non-overlapping cells, then for the non-overlapping cases analyses the shape and extracts a series of measurements, including the number of "corner" or pointy edges through a multiscale angle variation matrix, *anglegram*. The shape evolution algorithm was tested on fluorescently labelled macrophages observed on embryos of *Drosophila melanogaster*.

Index Terms— Shape evolution, active contours

1. INTRODUCTION

Cell migration processes are essential in numerous areas, for example in homoeostasis, where macrophages hold roles that range from repairing tissue through to immune responses to pathogens [1]. Macrophages are cells of the immune system that filter microbes and foreign particles when settled in lymphoid tissues and the liver [2], these processes imply sophisticated migration mechanisms. The use of *Drosophila Melanogaster* as a model organism has led to insights about how the macrophages integrate cues to migration [3].

A precise analysis of the cell shapes as they evolve through time could provide information for specific cells in biological studies, where sharp corners suggest an active migrating cell and rounded corners inactivity. The interactions amongst the cells' structures appear to anticipate migration [4] and different shape states could hint different movements. Such an analysis would require an accurate segmentation that incorporates shape information as it evolves through time. Segmentation of single frames and in the temporal context is a widely studied area [5]. Single frame segmentation of fluorescent microscopy images involves a broad variety of techniques, like automatic thresholding [6], curve evolution techniques such as active contours, of active surfaces [7].

On the other hand, tracking techniques have been used to follow link segmented cells into a track. Some techniques include the keyhole algorithm [8] or more sophisticated techniques that optimise both the tracking and the segmentation in conjunction [9]. In this work a framework to segment, track and analyse the shape of migrating macrophages is described.

2. MATERIALS AND METHODS

2.1. Macrophages embryos

Five hundred and forty one time frames with dimensions $(n_h, n_w, n_d) = (512, 672, 3)$ and two fluorescent channels (red = GFP-Moesin, green = Clip-GFP) were acquired as described in [4]. Figure 2 shows the temporal variation of one cell. Four basic shapes, Fig. 1, can be recognised in the data: **circles or ellipses, drops** (one pointy edge like a water drop), **bi-drops** and **tri-drops** (similar to a drop but with more than one pointy edge).

(a) Full frame.   (b) Regions of interest with basic shapes

Fig. 1: One representative time frame where all basic shapes can be recognised. The full frame is displayed in (a) with (red) squares highlight the cells of interest. The detail of each cell is shown in (b).

2.2. Synthetic data

A collection of 1000 images of basic shapes was generated with control points joined by cubic splines whose independent variable ranged in $\tau \in [0, 2\pi]$. Let $\{Y_i^\star\}_1^N = \{(x_{1,i}, y_{2,i})\}_1^N$, be a collection of N control points of a

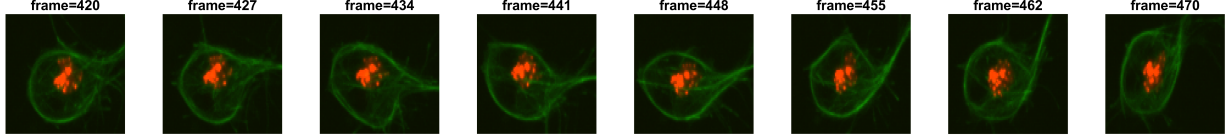| frame=420 | frame=427 | frame=434 | frame=441 | frame=448 | frame=455 | frame=462 | frame=470 |

Fig. 2: Evolution of a cell shape in time. The cell is shown in 8 different instances from 50 consecutive time frames.

basic shape such that $Y_i^\star \sim \mathcal{N}(\mu_i^\star, \sigma_i^\star)$. The value of N depends on the type of basic shape: the **circle** has 4, **drop** has 7, **bi-drop** has 8 and **tridrop** has 10. To model the variations in the cells' shapes, within their basic categories, the control points are distributed Normal. The control points are joined with a spline that then produces the **boundary** of the shape, $\mathcal{B}$, which then models that of a segmented cell (Fig. 3).
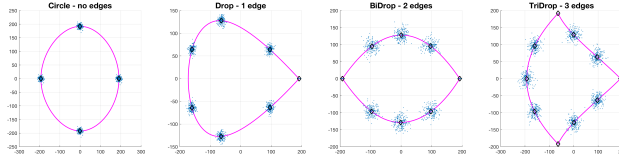

Fig. 3: Synthetic generation of random basic shapes. Per shape, 200 cases were generated. The control points are shown in blue($\cdot$). The mean shapes are presented in magenta($-$); and the mean control points are represented in black($\diamond$).

2.3. Shape tracking methodology

The shape of the cells was extracted from the green channel whilst the red channel was used to distinguish between individual and overlapping cells. Both channels were low pass filtered with a 5×5 Gaussian kernel and then segmented by Otsu thresholding [10]. Finally, a morphological opening with a disk structural element ($r = 3$) was performed to smooth the edges and remove noise. As reported in [11], this segmentation can be used to distinguish between overlapping groups of cells called *clumps*. This work is focused in tracking the non overlapping cells. The main functionality presented is a framework in which a cell shape can be tracked through a curve evolution algorithm. The PhagoSight package [8] was used to track the movement of the nuclei through time using the keyhole algorithm, and the segmentation of the green channel was used to determine which nuclei were involved in a clump. Clumps were removed from the analysis. The tracking produced unique labels for each of the tracks detected, as well as the position of each nuclei at each point in time. Each track is represented by $\mathcal{T}_j$, with $j = 1, 2, \cdots, M$ being the number of tracks. Each $\mathcal{T}_j$ contained the information regard-

ing the positions of the centroid of the nuclei $\{\mathbf{x}_{j,k}\}_{k=1}^{T}$ as well as the time frames where the track was present $t_{j,0}, \cdots, t_{j,k}, \cdots, t_{j_0, T-1}$ for the T consecutive time frames. For simplicity, a time frame of an arbitrary track $\mathcal{T}$ is shown as t_k.

Given a track $\mathcal{T}$, the shape of a cell $\mathcal{B}_{k+1}$ at any time t_{k+1} can be determined by the shape of the previous frame $\mathcal{B}_k$, and the position change from t_k into t_{k+1} of the red nuclei. Let $\bar{\mathcal{B}}_k$ be the shape $\mathcal{B}_k$ when moved onto the position $\mathbf{x}_{k+1}$. The shape $\mathcal{B}_{k+1}$ will be determined by **evolving** $\bar{\mathcal{B}}_k$ the shape from the previous time frame, where $\mathbb{I}_k$ corresponds to the image of the k_{th} time frame (Algorithm 1). Every iteration of the algorithm involves loading the known frame t_k, the position information from the unknown frame t_{k+1}, performing the evolution from $\bar{\mathcal{B}}_{k+1}$ to $\mathcal{B}_{k+1}$ and obtaining the region properties of the new cell shape, these include (i) the *orientation*, (ii) the ratio of the minor and major axes (*aspect ratio*), (iii) the *solidity*, and (iv) the *equivalent diameter*. The **evolve** function in Algorithm 1 is an implementa-

---

Algorithm 1: SHAPE EVOLUTION Tracks shape of cells in a single track.

   Input: Track: $\mathcal{T}$, time frames: $(t_0 : t_{T-1})$
1 $t_k \leftarrow t_0$;
2 $(\mathbb{I}_k, \mathbf{x}_k)$load frame information at t_k;
3 $\mathcal{B}_k \leftarrow$ get boundary($\mathbb{I}_k$);
4 **for** t_{k+1} *in time frames* **do**
5 $t_{k+1} \leftarrow t_k + 1$;
      $(\mathbb{I}_{k+1}, \mathbf{x}_{k+1}) \leftarrow$ load frame at t_{k+1};
      $d_k \leftarrow \mathbf{x}_{k+1} - \mathbf{x}_k$;
6 $\bar{\mathcal{B}}_{k+1} \leftarrow$ move boundary($\mathcal{B}_k, d_K$);
7 $\mathcal{B}_{k+1} \leftarrow$ evolve($\mathbb{I}_{k+1}, \bar{\mathcal{B}}_{k+1}$);
8   save($\mathcal{B}_{k+1}$, regionprops($\mathcal{B}_{k+1}$));
9 $t_k \leftarrow t_{k+1}$;
10 $\mathcal{B}_k \leftarrow \mathcal{B}_{k+1}$;
11 **end for**

---

tion of the Chan-Vese active contour [12, 13] method in MATLAB®. The function uses the moved boundary $\bar{\mathcal{B}}$ as initialisation and is able to change its parameters based on one of three states: Shrink, Grow, or Normal (Table

2

1). The active contour is run once, with a set of parameters, then the area of the output is compared to the area of the input. The parameters would be adjusted to contract or expand the shape and the active contour is run again. In order to avoid an excessive leaking or contraction of the segmentation, the area of the output was kept within $\pm 5\%$ of the previous frame's area.

Table 1: Parameters used of the active contour function based on the desired state required. The parameters were chosen empirically through numerous tests.

State	Iterations	smooth factor	contraction bias
Normal	50	1.5	-0.1
Shrink	100	1.25	0.10
Grow	200	1.00	-0.25

2.4. Shape analysis through *Anglegram*

The *anglegram* [11] is a matrix, which describes multiscale angle variation of a shape. For each one of the ordered points $\mathbf{p}_i \in \mathcal{B}$, the inner angles of every point is computed for every separation and the maximum intensity projection (*MIP*) per columns is calculated. To detect clumps the angles of interest were those larger than one standard deviation (*std*) above the mean of the MIP. In this work, a similar idea is explored, but for *acute* angles, using the minimum intensity projection (*mIP*) and the threshold is one *std* below the mean. The implementation of the anglegram matrix involves the detection of junctions that correspond to the corners of the analysed boundary. Figure 5 shows the junction detection functionality of the anglegram. The method works as reported in [11], with two alterations: (i) resizing the anglegram to have 64 rows to reduce noise, and (ii) taking the *minimum intensity projection* of the first half columns, as the final columns are lower by the definition of the inner point angle measurements. The corner detection method was tested on 70,000 randomly generated boundaries of the different types of shapes. Then, the corners were detected on the shapes of the cells being tracked for a qualitative assessment.
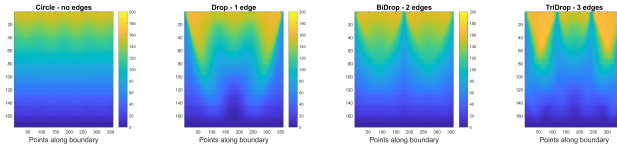
Fig. 4: Anglegram on mean basic shapes from Figure 3. The anglegram matrix was calculated and transposed for visualisation purposes. From left to right: **circle, drop, bidrop** and **tridrop**.

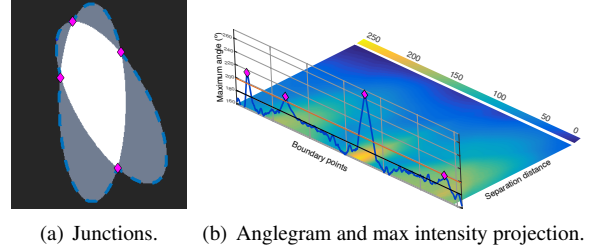(a) Junctions.   (b) Anglegram and max intensity projection.

Fig. 5: Description of the original junction detection by *anglegram*. The junctions detected on a synthetic pair or ellipses is shown in (a), where the boundary of the overlap is represented in blue(–) and the junctions in magenta ($\diamond$). In (b), the anglegram matrix is displayed in a plane and the *MIP* is represented along the boundary points. Detection of junctions are shown in magenta ($\diamond$).

3. RESULTS

Tests on the synthetic data were in the range of 86-95% correct detection (Table 2). The method achieved the highest accuracy in the **bidrop** shape, and the lowest in the **drop** shape. Figure 6 displays the shape evolution

Table 2: Number of corners detected per shape. The accuracy of the method is highlighted.

Shapes /	Number of corners detected (%)				
Corners	None	1	2	3	$\geqslant 4$
Circle	**89.35**	0	0	0.47	10.18
Drop	4.42	**86.02**	6.81	2.52	0.23
Bidrop	0	0.41	**95.17**	4.31	0.11
Tridrop	0	0.02	2.76	**92.94**	4.28

of a track-fragment containing 50 frames in the dataset. Eight frames are shown with the moved boundary displayed in the dotted line (cyan- -) and the evolved shape in the solid line (magenta-). The corners detected by the anglegram at each point are also displayed as asterisks (yellow∗) for qualitative assessment. The orientation and aspect ratio are plot together in the middle row, in the bottom row, the number of corners and the measured angle.

4. DISCUSSION

Preliminary work involving curve evolution methods such as multilevel set methods [14] and active contours [12] have proven unsuccessful to segment even non overlapping cells in the analysis of a single frame. Such techniques require considerable precision when selecting the parameters, but especially when selecting the initialisation given to the algorithms. In this work, a method
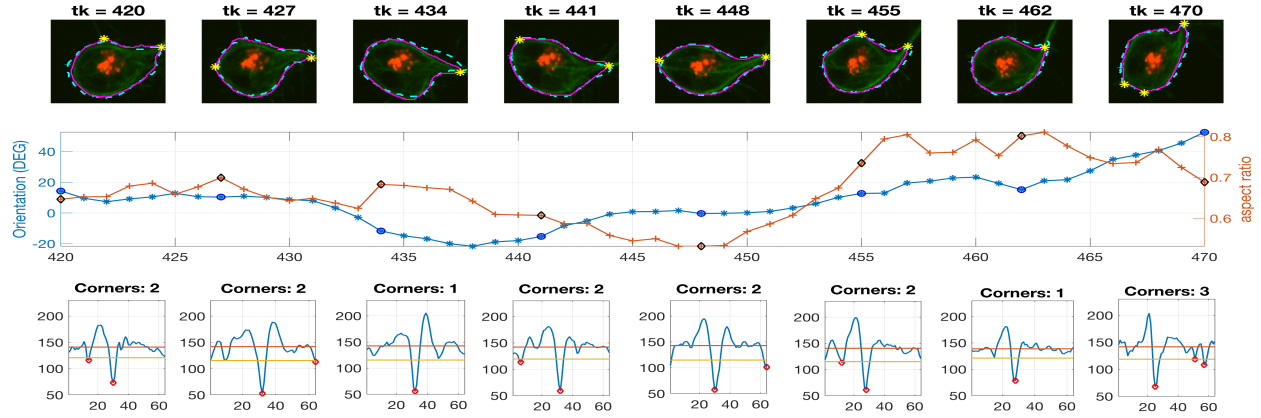
3

Fig. 6: Evolution of cell shape throughout multiple frames. **Top row**: Eight instances of 50 consecutive frames. Previous segmentation in cyan(- -), evolved in magenta(-) and corners detected in yellow(∗). **Middle row**: Comparison between the orientation (blue ∗−) and the aspect ratio (orange +−). The values for the **top** frames are highlighted in blue(○) and black (◇). **Bottom row**: Minimum intensity projection of **top**'s' respective anglegrams displaying the values of the angles measured per corner (red ◇).

to segment moving cells based on active contours was proposed.

The main contribution was to provide a framework for the consistent tracking of the shape of a cell and provide a measured evolution of some shape parameters. As shown in Figure 6, the similarity of the previous frame to the current is enough to provide an appropriate initialisation for the active contours, however, as shown in Table 1, monitoring of the shape is necessary to avoid segmentation leaking. A new implementation of the *anglegram* matrix allowed for the analysis of a single cell with a straightforward identification of corners in the shapes. Future developments involve extending the shape tracking into overlapping cells to disambiguate them. Furthermore, as seen in Figure 4, the distinctive patterns of the anglegrams corresponding to the basic shapes, could be further explored into a classification problem for shape identification or correction of irregular segmentations.

5. REFERENCES

[1] T Wynn *et al.*, "Macrophage biology in development, homeostasis and disease.," *Nature*, vol. 496, no. 7446, pp. 445–55, apr 2013.

[2] F Martinez *et al.*, "Macrophage activation and polarization," *Front Biosci*, vol. 13, pp. 453—461, 2008.

[3] W Wood *et al.*, "Macrophage functions in tissue patterning and disease: New insights from the fly," *Developmental Cell*, vol. 40, no. 3, pp. 221–233, 2017.

[4] B Stramer *et al.*, "Clasp-mediated microtubule bundling regulates persistent motility and contact repulsion in Drosophila macrophages in vivo.," *J Cell Biology*, vol. 189, no. 4, pp. 681–9, 2010.

[5] M Maška *et al.*, "A benchmark for comparison of cell tracking algorithms," *Bioinformatics*, vol. 30, no. 11, pp. 1609–1617, 2014.

[6] I Hannah *et al.*, "The use of variance and entropic thresholding methods for image segmentation," *Patt Rec*, vol. 28, no. 8, pp. 1135–1143, 1995.

[7] A Dufour *et al.*, "Segmenting and tracking fluorescent cells in dynamic 3-d microscopy with coupled active surfaces," *IEEE Trans Imag Proc*, vol. 14, no. 9, pp. 1396–1410, 2005.

[8] K Henry *et al.*, "PhagoSight: An Open-Source MATLAB® Package for the Analysis of Fluorescent Neutrophil and Macrophage Migration in a Zebrafish Model," *PLoS ONE*, vol. 8, no. 8, pp. e72636, 2013.

[9] R Collins *et al.*, "Hybrid stochastic / deterministic optimization for tracking sports players and pedestrians," in *Computer Vision – ECCV*, Cham, 2014, LNCV, pp. 298–313, Springer.

[10] N Otsu , "A Threshold Selection Method from Gray-Level Histograms," *IEEE Trans Sys, Man, and Cyber*, vol. 9, no. 1, pp. 62–66, 1979.

[11] J Solís-Lemus *et al.*, "Segmentation of Overlapping Macrophages Using Anglegram Analysis," in *Medical Image Understanding and Analysis*. July 2017, CCIS, Springer, Cham.

[12] T Chan *et al.*, "Active contours without edges.," *IEEE Trans Imag Proc*, vol. 10, no. 2, pp. 266–277, jan 2001.

[13] R Whitaker , "A level-set approach to 3d reconstruction from range data," *Int J Computer Vision*, vol. 29, no. 3, pp. 203–231, Sep 1998.

[14] Z Lu *et al.*, "An improved joint optimization of multiple level set functions for the segmentation of overlapping cervical cells," *IEEE Trans Imag Proc*, vol. 24, no. 4, pp. 1261–1272, 2015.