

Msc. Seol Ah Park

Dissertation Thesis Abstract

Tracking cells in 2D +time and 3D + time image sequences

to obtain the Academic Title of *philosophiae doctor (PhD.)*
in the doctorate degree study programme
9.1.9 Applied Mathematics
full-time study

Dissertation Thesis has been prepared at the Department of Mathematics and Descriptive Geometry, Faculty of Civil Engineering, Slovak University of Technology in Bratislava.

Submitter: Msc. Seol Ah Park
Department of Mathematics and Descriptive Geometry
Faculty of Civil Engineering, STU, Bratislava

Supervisor: Prof. RNDr. Karol Mikula, DrSc.
Department of Mathematics and Descriptive Geometry
Faculty of Civil Engineering, STU, Bratislava

Reviewers: doc. RNDr. Angela Handlovičová, PhD.
Department of Mathematics and Descriptive Geometry
Faculty of Civil Engineering, STU, Bratislava

Ing. Jozef Urbán, PhD.
TatraMed Software s.r.o
Bratislava

prof. RNDr. Michal Fečkan, DrSc.
Department of Mathematical Analysis and Numerical Mathematics
Faculty of Mathematics, Physics and Informatics, Comenius University in Bratislava

Dissertation Thesis Abstract was sent:

Dissertation Thesis Defence will be held on at am/pm at Department of Mathematics and Descriptive Geometry, Faculty of Civil Engineering, Slovak University of Technology in Bratislava, Radlinskeho 11.

prof. Ing. Stanislav Unčík, PhD.
Dean of Faculty of Civil Engineering

Abstract

In this thesis, we develop methods of image segmentation and cell tracking, especially for cells that migrate and change their shapes dynamically. The studied methods are applied to biological time-lapse images, mostly for macrophages known as one of the fastest-moving cellular populations with complex shapes and movements. For the first step in image segmentation, we design a segmentation method combining thresholding techniques and the SUBSURF (subjective surface segmentation) method. Before applying the combination of thresholding and SUBSURF, space-time filtering, which can preserve temporal coherence in time-lapse data is performed. The presented segmentation method can capture cells that have a wide range of image intensities and shapes. The accuracy of the segmentation is measured by using the Hausdorff distance, the IoU (Jaccard) index, and the Sørensen–Dice coefficient between the semi-automatic and automatic segmentation. In the segmented images, the approximate cell centers in every time frame are found by solving the time-relaxed eikonal equation. In the next step, the partial trajectories for cells overlapping in the temporal direction are extracted. Then, the tangent calculation at endpoint points of partial trajectories is carried out to link the trajectories for non-overlapping cells. The accuracy of the cell tracking method is evaluated by counting the number of correct links, and the trajectories obtained from the proposed method are compared with the manually extracted trajectories by using the mean Hausdorff distance.

Keywords: Image segmentation, global thresholding, adaptive thresholding, subjective surface method, finite volume method, semi-implicit scheme, cell tracking, time-relaxed eikonal equation, macrophages.

Abstrakt

V tejto práci vyvíjame metódy segmentácie obrazu a trekingu buniek, a to najmä pre bunky, ktoré dynamicky migrujú a menia svoj tvar. Navrhované metódy sú aplikované na biologické časopriestorové videá, najmä na treking makrofágov, ktoré sú známe ako jedna z najrýchlejších sa pohybujúcich bunkových populácií so zložitými tvarmi buniek a ich zmenami. Na segmentáciu obrazu navrhujeme kombináciu prahovania a metódy subjektívnych plôch (SUBSURF), ktorým predchádza časopriestorová filtrácia videa, zachovávajúca časovú koherenciu objektov v časopriestorových dátach. Prezentovaná segmentačná metóda dokáže zachytiť bunky, ktoré majú širokú škálu intenzít a tvarov. Presnosť segmentácie sa meria pomocou Hausdorffovej vzdialenosti medzi semi-automatickou a automatickou segmentáciou. Vo vysegmentovaných obrazoch sa následne v každej časovej snímke nájdu približné stredy buniek pomocou riešenia časovo závislej eikonalovej rovnice. V ďalšom kroku sa extrahujú čiastkové trajektórie pre bunky prekrývajúce sa v čase. Potom sa vykoná výpočet dotyčníc v koncových bodoch čiastkových trajektórií, aby sa prepojili trajektórie pre neprekrývajúce sa bunky. Presnosť metódy trekingu buniek sa vyhodnocuje určením počtu správnych prepojení v každom časovom kroku ako aj pomocou strednej Hausdorffovej vzdialenosti medzi trajektóriami získanými navrhovanou metódou a manuálne extrahovanými trajektóriami.

Kľúčové slová: Segmentácia obrazu, globálne prahovanie, adaptívne prahovanie, metóda subjektívnych plôch, metóda konečných objemov, semi-implicitná schéma, treking buniek, eikonalá rovnica, makrofágy.

Contents

1	Introduction	3
2	Image segmentation	4
2.1	Space-time filtering	5
2.2	The local Otsu's method	7
2.3	The subjective surface segmentation method	8
2.4	Experiments with 2D time-lapse data of macrophages	9
3	Cell tracking	12
3.1	Detection of the approximate cell center	12
3.2	Extraction of partial trajectories	13
3.3	Estimation of cell movement direction	15
3.4	Connection of partial trajectories	17
3.5	Experiments with 2D time-lapse data of macrophages	18
	List of publication	21
	References	22

1. Introduction

The development of computer vision has contributed significantly to the analysis of biological data by providing automated tools such as the segmentation and tracking of cells. In biology, there have been many efforts to understand cellular or subcellular dynamics in living animals by using images acquired by microscopes. The automatic segmentation and tracking of individual cells are the first tasks to analyze cell behaviors objectively.

Identifying (segmentation) and tracking individual cells is challenging because cells divide, move, and change their shapes during their journey in the developing embryo. A lot of efforts have been dedicated to developing software to track cells during embryonic development and robust solutions are now available [1]. Some of these solutions are even compatible with the study of other situations where cells are either moving in an organism (the heart) or in a moving organism (neurons in foraging worms [2]) but some specific cellular populations, due to their very specific behaviors are difficult to identify and track during their journey within a living animal. This is the case with macrophages which are one of the fastest-moving cellular populations with more erratic shapes and movements.

Segmentation of macrophages has been previously studied performing a filter-based method [3], image-based machine learning [4], anglegram analysis [5], etc. Also, deep learning-based segmentation methods have been developed for various types of cells [6, 7, 8, 9, 10, 11]. U-Net[6, 8], Cellpose[10], and Splinedist[11] are designed for segmentation of general shapes of cells in microscopy data and have shown a high performance. However, it is still a challenging task to segment macrophages due to their varying nature and extreme irregularity of shapes, and variability of image intensity inside macrophages. In [12], we have proposed a macrophage segmentation method that combines thresholding methods with the SUBSURF method requiring no cell nuclei center or other reference information. However, a problem occurs when attempting to segment macrophages in time-lapse data, since the segmentation parameters are not always suitable for macrophages in all time frames. In this thesis, first, we improve the ability to detect macrophages with low image intensity by applying space-time filtering which considers the temporal coherence of time-lapse data [13]. Second, Otsu's method is implemented in local windows to deal with cases in which each macrophage has a substantially different image intensity range. Similarly, as in [12], the SUBSURF method [14] is applied to eliminate the remaining noise and to smoothen the boundaries of the macrophages resulting from space-time filtering and the thresholding method (Figure 1.1). The performance of the proposed image segmentation is presented by computed mean Hausdorff distances between the results of the semi-automatic method, the gold standard in this thesis. Also, the perimeter, area, and circularity are compared between the proposed method and other segmentation methods. This comparison shows the proposed method is the most similar to the results of the gold standard.

Automatic cell tracking in microscopy images has been investigated and several different methods [15, 16, 17, 18, 19, 20, 21, 9, 22] have been proposed. The tracking algorithm by using linear assignment problem (LAP) [15, 21] is computationally efficient and has shown good performance, especially for Brownian motion. However, it can be less accurate if a large number of cells is densely distributed or if some cells suddenly move toward the other nearby cells. The studies [18, 20] performed the cell tracking during the zebrafish embryogenesis by finding a centered path in the spatio-temporal segmented structure. In [19], a workflow was designed, from the image acquisition to cell tracking, and applied to 3D+time microscopy data of the zebrafish embryos. Those methods show outstanding performance in the case of embryogenesis. Meanwhile, the keyhole tracking algorithm has proposed and been applied to red blood cells, neutrophils, and macrophages [23, 24, 25]. Furthermore, deep learning-based motion tracking in microscopy images has been studied for various types of biological objects with different learning approaches [26, 27, 28, 29, 30, 31, 32, 33, 34, 35]. For instance, the method in [28] trains the networks by utilizing semi-supervised learning to predict cell division. Usiigaci [30] segments individual cells providing each unique ID to them with a Mask R-CNN model, then the method links the cells by given IDs. The methods by training image sequences using LSTM (long short-term memory) networks have shown their performance for tracing nuclear proteins [31] and bacteria [33]. In [32], the algorithm to solve linear assignment problems in tracking is trained with a deep reinforcement learning (DRL)-based

method. Although various methods of cell tracking have been studied, there is still a need for more accurate tracking of erratic movements, such as macrophages.

The cell tracking studied in this thesis deals with macrophages which undergo fast and complicated motion. It results in non-overlapping cells in the time direction and, in many cases, one can observe a “random movement”. This paper proposes a tracking method that covers the situations of a large number of macrophages and their complex motion. The first step is to extract the cell trajectories from their shapes overlapping in time. By this approach, we often extract only partial trajectories, because not always a segmented macrophage overlaps with its corresponding cell in the next/previous frame of the video. Next, we connect endpoints of partial trajectories corresponding to macrophages that do not overlap in time. To do this, the tangent calculation is used to estimate the direction of the partial trajectories at the endpoints. Figure 1.1 illustrates briefly all steps of the proposed method yielding macrophage tracking. The performance of the tracking method is analyzed by measuring the mean Hausdorff distance between trajectories obtained from manual and automatic tracking. In this measurement. Also, The mean accuracy of tracking is checked by averaging the ratio between the correct links and total links over all time slices. The proposed tracking method showed a high mean accuracy of 97.4%.

The remaining part of the thesis is organized as follows. In Chapter 2, the proposed segmentation method will be illustrated through the four subsections. The first subsection will explain space-time filtering as pre-smoothing. The second and third subsections describe the local Otsu’s method, one type of local thresholding technique, and the SUBSURF method. The last subsection of this chapter will propose the combination of the threshold and SUBSURF method and apply the method to 2D+time macrophage data. Chapter 3 will present the proposed algorithm of cell tracking with two steps. The first subsection will illustrate the time-relaxed eikonal equation to approximate the center of the cells. The second subsection will show the algorithm for the extraction of partial trajectories. Then, the connection of those partial trajectories will be described in the third subsection of this chapter. The application in 2D+time and its quantitative comparison will be presented in the last subsection.

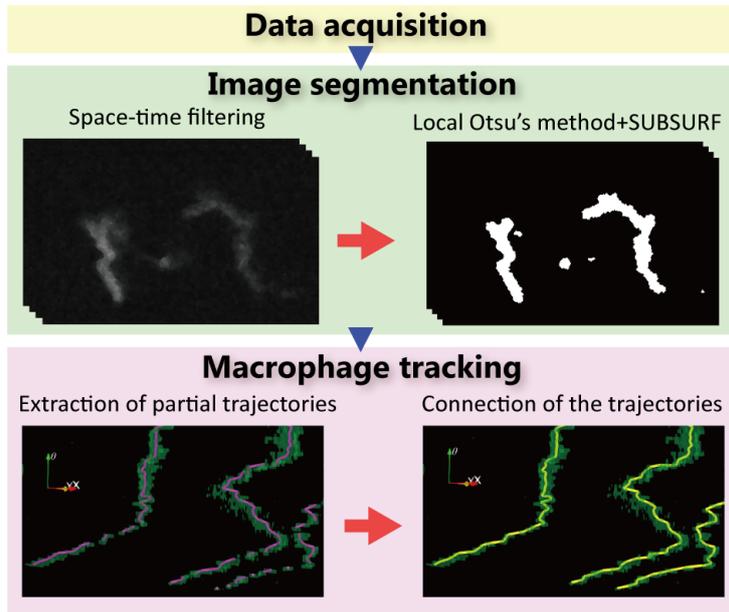


Figure 1.1: Procedure of macrophage tracking in 2D+time data.

2. Image segmentation

For image segmentation, we design the method by combining threshold techniques and the SUBSURF method. Before performing it, the time-lapse images are processed by using space-time filtering. This filtering can keep the temporal coherence of objects so that making distinguishable the signals between objects and the background noise. With the images obtained by space-time filtering, the filtered images are binarized by

applying a threshold technique, the local Otsu's method. The local Otsu's method enables us to capture the objects having high variability of image intensity and shapes. Finally, the SUBSURF method removes the artifact and remaining detected noise in the images obtained from the local Otsu's method, and also smoothes the boundary of objects.

2.1 Space-time filtering

The image sequence after space-time filtering can be represented by a real function $u(t, \mathbf{x}, \theta)$ defined on $[0, T] \times \Omega \times [0, \theta_F]$, where t denotes the scale, in other words the amount of filtering, \mathbf{x} is a spatial domain, $\mathbf{x} \in \Omega \subset \mathbb{R}^N$, and θ denotes a particular time slice with the interval $[0, \theta_F]$.

The equation of space-time filtering is represented by the following nonlinear diffusion,

$$\frac{\partial u}{\partial t} = clt(u) \nabla \cdot (g(|\nabla G_\sigma * u|) \nabla u) \quad (2.1)$$

with the initial condition given by

$$u(0, \mathbf{x}, \theta) = u^0(\mathbf{x}, \theta). \quad (2.2)$$

In $|\nabla G_\sigma * u|$, the “*” stands for the convolution operator. The $clt(u)$ function is defined as in [36, 13] by formula

$$clt(u) = \min_{w_1, w_2} \frac{1}{(\Delta\theta)^2} (|\langle \nabla u, w_1 - w_2 \rangle| + |u(\mathbf{x} - w_1, \theta - \Delta\theta) - u(\mathbf{x}, \theta)| + |u(\mathbf{x} + w_2, \theta + \Delta\theta) - u(\mathbf{x}, \theta)|), \quad (2.3)$$

where w_1, w_2 are arbitrary vectors in the space and $\Delta\theta$ is the time increment between discrete time slices. The function g is the so-called edge detector function and is defined by

$$g(s) = \frac{1}{1 + Ks^2}, \quad K > 0. \quad (2.4)$$

Finally, G_σ is a Gaussian function with variance σ which is used for pre-smoothing by convolution. Let us denote by u_k^n a numerical solution in the k^{th} frame of the image sequence in the n^{th} discrete filtering (scale) step $n\tau_F$ with the step size τ_F , i.e.

$$u_k^n(\mathbf{x}) = u(n\tau_F, \mathbf{x}, k\Delta\theta). \quad (2.5)$$

By using the semi-implicit scheme [13], Equation (2.1) is discretized as follows

$$\frac{u_k^{n+1} - u_k^n}{\tau_F} = clt(u_k^n) \nabla \cdot (g(|\nabla u_k^{\sigma;n}|) \nabla u_k^{n+1}), \quad (2.6)$$

where $g(|\nabla u_k^{\sigma;n}|) = g(|\nabla G_\sigma * u_k^n|)$. From Equation (2.3), the discretization of $clt(u_k^n)$ in the point $\mathbf{x} \in \Omega$ can be written as

$$clt(u_k^n) = \min_{w_1, w_2} \frac{1}{\Delta\theta^2} (|\langle \nabla u_k^n, w_1 - w_2 \rangle| + |u_{k-1}^n(\mathbf{x} - w_1) - u_k^n(\mathbf{x})| + |u_{k+1}^n(\mathbf{x} + w_2) - u_k^n(\mathbf{x})|), \quad (2.7)$$

For space discretization, we use the finite volume method with finite volume (pixel) side h . By considering that a point v is a center of a pixel (i, j) and let us denote by \mathcal{V}_{ij} a finite volume corresponding to pixel (i, j) , $i = 1, \dots, M$, $j = 1, \dots, N$. The quantity $clt(u_k^n)$ is considered constant in finite volumes. Then, Equation (2.6) is integrated with the finite volume \mathcal{V}_{ij} and by using Green's theorem we get

$$\int_{\mathcal{V}_{ij}} \frac{u_k^{n+1} - u_k^n}{\tau_F} d\mathbf{x} = clt(u_k^n) \int_{\partial\mathcal{V}_{ij}} g(|\nabla u_k^{\sigma;n}|) \nabla u_k^{n+1} \cdot \mathbf{n}_{ij} dS, \quad (2.8)$$

where \mathbf{n}_{ij} is a unit outward normal vector to the boundary of \mathcal{V}_{ij} . The gradient of u on the pixel edges can be approximated by computing the average values of neighboring pixels. By using the diamond cell approach [37], we compute the average of neighboring pixel values in the corners of the pixel (i, j) as follows (see also Figure 2.1).

$$\begin{aligned}
u_{i,j,k}^{1,1} &= \frac{1}{4}(u_{i,j,k}^n + u_{i,j+1,k}^n + u_{i+1,j,k}^n + u_{i+1,j+1,k}^n), \\
u_{i,j,k}^{1,-1} &= \frac{1}{4}(u_{i,j,k}^n + u_{i+1,j,k}^n + u_{i,j-1,k}^n + u_{i+1,j-1,k}^n), \\
u_{i,j,k}^{-1,-1} &= \frac{1}{4}(u_{i,j,k}^n + u_{i-1,j,k}^n + u_{i,j-1,k}^n + u_{i-1,j-1,k}^n), \\
u_{i,j,k}^{-1,1} &= \frac{1}{4}(u_{i,j,k}^n + u_{i,j+1,k}^n + u_{i-1,j,k}^n + u_{i-1,j+1,k}^n).
\end{aligned} \tag{2.9}$$

The gradient of $u_{i,j,k}^n$ in n^{th} filtering step, for a pixel (i, j) in k^{th} frame of the image sequence, is computed at the center of edges of the pixel [37],

$$\begin{aligned}
\nabla^{1,0} u_{i,j,k}^n &= \frac{1}{h}(u_{i+1,j,k}^n - u_{i,j,k}^n, u_{i,j,k}^{1,1} - u_{i,j,k}^{1,-1}), \\
\nabla^{0,-1} u_{i,j,k}^n &= \frac{1}{h}(u_{i,j,k}^{1,-1} - u_{i,j,k}^{-1,-1}, u_{i,j-1,k}^n - u_{i,j,k}^n), \\
\nabla^{-1,0} u_{i,j,k}^n &= \frac{1}{h}(u_{i-1,j,k}^n - u_{i,j,k}^n, u_{i,j,k}^{-1,1} - u_{i,j,k}^{-1,-1}), \\
\nabla^{0,1} u_{i,j,k}^n &= \frac{1}{h}(u_{i,j,k}^{1,1} - u_{i,j,k}^{-1,1}, u_{i,j+1,k}^n - u_{i,j,k}^n),
\end{aligned} \tag{2.10}$$

where h denotes the pixel size. With the set of grid neighbors $N_{i,j}$ that consists of all (l, m) of $\mathcal{V}_{i,j}$, such that $l, m \in \{-1, 0, 1\}$, $|l| + |m| = 1$, the final discretized form of equation (2.1) is written as

$$u_{i,j,k}^{n+1} = u_{i,j,k}^n + \frac{\tau_F}{h^2} \text{clt}(u_{i,j,k}^n) \sum_{|l|+|m|=1} g(|\nabla^{l,m} u_{i,j,k}^n|) (u_{i+l,j+m,k}^{n+1} - u_{i,j,k}^{n+1}). \tag{2.11}$$

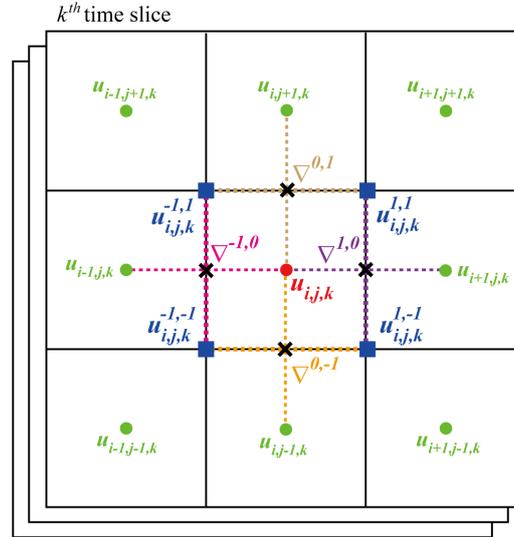


Figure 2.1: Schematic picture of the space discretization. Here, $u_{i,j,k}$ denotes the value of the unknown function u at a pixel position (i, j) in k^{th} time slice. In a clockwise direction, the values of u at corners, $u_{i,j,k}^{1,1}$, $u_{i,j,k}^{1,-1}$, $u_{i,j,k}^{-1,-1}$, and $u_{i,j,k}^{-1,1}$, are defined. The gradients, $\nabla^{1,0}$, $\nabla^{0,-1}$, $\nabla^{-1,0}$, and $\nabla^{0,1}$, are computed at the center of edges.

2.2 The local Otsu's method

The global threshold method described above works well if there is only one object in the image or the range of image intensity is similar between objects. However, it is needed that the image thresholds are inspected locally when objects have huge variability of image intensity as mentioned at the beginning of the chapter. In this section, we calculate the optimal thresholds for every pixel in a certain window of size $s \times s$ centered in a pixel (i, j) by using the approach of the global Otsu's method.

In the local window W_{ij} , the gray-level histogram is normalized and a probability distribution is regarded as

$$p_r = n_r/N, \quad \sum_{r=0}^L p_r = 1, \quad (2.12)$$

where n_r is the number of pixels of intensity r in W_{ij} , $N = s^2$ and L is the maximum image intensity. Then, the probabilities of background and foreground in W_{ij} are given by

$$\omega_0(T_{ij}) = \sum_{r=0}^{T_{ij}} p_r, \quad \omega_1(T_{ij}) = \sum_{r=T_{ij}+1}^L p_r = 1 - \omega_0(T_{ij}) \quad (2.13)$$

and means of background and foreground are given by

$$\begin{aligned} \mu_0(T_{ij}) &= \frac{1}{\omega_0(T_{ij})} \sum_{r=0}^{T_{ij}} r p_r, \\ \mu_1(T_{ij}) &= \frac{1}{\omega_1(T_{ij})} \sum_{r=T_{ij}+1}^L r p_r = \frac{\mu_{tot} - \mu_0(T_{ij})\omega_0(T_{ij})}{1 - \omega_0(T_{ij})}, \end{aligned} \quad (2.14)$$

where $\mu_{tot} = \sum_{r=0}^L r p_r$. Finally, the between-class variance related to the pixel (i, j) is written as

$$\sigma_B^2(T_{ij}) = \omega_0(T_{ij})(\mu_0(T_{ij}) - \mu_{tot})^2 + \omega_1(T_{ij})(\mu_1(T_{ij}) - \mu_{tot})^2 \quad (2.15)$$

which simplifies to

$$\sigma_B^2(T_{ij}) = \frac{(\mu_{tot}\omega_0(T_{ij}) - \mu_0(T_{ij})\omega_0(T_{ij}))^2}{\omega_0(T_{ij})(1 - \omega_0(T_{ij}))}, \quad (2.16)$$

and the optimal threshold T_{ij}^* is given by

$$\sigma_B^2(T_{ij}^*) = \max_{0 \leq T_{ij} < L} \sigma_B^2(T_{ij}). \quad (2.17)$$

At the boundary of the image, mirroring is applied. In the case where the local window contains only the background, the histogram completely loses its bi-modality, with the threshold T_{ij}^* representing some noise level. To obtain a reasonable threshold, we determine whether the local window is only located in the background or not by considering the relative difference between the mean levels of two classes that represent the object and the background. Let us consider that $\mu_0(T_{ij}^*)$ and $\mu_1(T_{ij}^*)$ are the mean levels of the background and the object, respectively, based on the threshold T_{ij}^* . If $|\mu_0(T_{ij}^*) - \mu_1(T_{ij}^*)| < \varepsilon_l$, ε_l is very small, then the two classes cannot be properly separated, and it is reasonable to conclude that the local window is located in the background. In other words, the local window W_{ij} is considered as including an object when the following condition is fulfilled:

$$\frac{|\mu_0(T_{ij}^*) - \mu_1(T_{ij}^*)|}{\mu_0(T_{ij}^*)} > \delta, \quad (2.18)$$

where the relative difference is considered because the background noise level is different in each time slice. Here, δ is a parameter to check whether the local window $W_{i,j}$ contains macrophages or not. If there is a part of macrophages in $W_{i,j}$, the relative difference in equation (2.18) will have a larger value than δ . Finally, the

binarized images are obtained

$$B(i, j) = \begin{cases} 1, & I(i, j) > T_{ij}^* \text{ and Equation (2.18) is fulfilled} \\ 0, & \text{otherwise} \end{cases} \quad (2.19)$$

where $I(i, j)$ is the image intensity of the pixel (i, j) and T_{ij}^* is given by Equation (2.17).

2.3 The subjective surface segmentation method

The SUBSURF method can effectively complete missing parts of boundaries, join adjacent level lines, and rapidly remove noise [14]. In particular, this method has previously been shown to be useful for segmenting macrophage data [12]. The SUBSURF method is described by

$$u_t = |\nabla u| \nabla \cdot \left(g \frac{\nabla u}{|\nabla u|} \right), \quad (2.20)$$

where u is a evolving level set function, and g is given by Equation (2.4) where $s = |\nabla I_\sigma^0|$ in Equation (2.20) and I_σ^0 is the original image pre-smoothed by the Gaussian kernel. The SUBSURF is applied independently to the images for every time frame. Therefore, we solve the unknown function $u(t, \mathbf{x})$, where $(t, \mathbf{x}) \in [0, T_S] \times \Omega$, $\mathbf{x} \in \Omega \subset R^N$, $N = 2, 3$. The time discretization of Equation (2.20) is given by the semi-implicit scheme

$$\frac{u^{n+1} - u^n}{\tau_S} = |\nabla u^n|_\varepsilon \nabla \cdot \left(g \frac{\nabla u^{n+1}}{|\nabla u^n|_\varepsilon} \right), \quad (2.21)$$

where τ_S is the scale step. Here, $|\nabla u^n|$ is regularized using the Evans–Sprucks approach [38] as $|\nabla u^n|_\varepsilon = \sqrt{|\nabla u^n|^2 + \varepsilon^2}$, where ε is a small arbitrary constant. The space is discretized by a finite volume square grid with a pixel side size of h . For \mathcal{V}_{ij} , Equation (2.20) is integrated, and using Green's formula we get

$$\int_{\mathcal{V}_{ij}} \frac{1}{|\nabla u^n|_\varepsilon} \frac{u^{n+1} - u^n}{\tau_S} d\mathbf{x} = \int_{\partial \mathcal{V}_{ij}} g \frac{\nabla u^{n+1}}{|\nabla u^n|_\varepsilon} \cdot \mathbf{n}_{ij} dS, \quad (2.22)$$

where \mathbf{n}_{ij} is a unit outward normal vector to the boundary of the pixel (i, j) . In a similar manner as in Equation (2.10), we use the diamond cell approach [37]. The average of neighboring pixel values in the four corners of the pixel (i, j) are written as

$$\begin{aligned} u_{i,j}^{1,1} &= \frac{1}{4}(u_{i,j}^n + u_{i,j+1}^n + u_{i+1,j}^n + u_{i+1,j+1}^n), \\ u_{i,j}^{1,-1} &= \frac{1}{4}(u_{i,j}^n + u_{i+1,j}^n + u_{i,j-1}^n + u_{i+1,j-1}^n), \\ u_{i,j}^{-1,-1} &= \frac{1}{4}(u_{i,j}^n + u_{i-1,j}^n + u_{i,j-1}^n + u_{i-1,j-1}^n), \\ u_{i,j}^{-1,1} &= \frac{1}{4}(u_{i,j}^n + u_{i,j+1}^n + u_{i-1,j}^n + u_{i-1,j+1}^n). \end{aligned} \quad (2.23)$$

The gradient $u_{i,j}^n$, in n^{th} filtering step for a pixel (i, j) , is approximated by

$$\begin{aligned} \nabla^{1,0} u_{i,j}^n &= \frac{1}{h}(u_{i+1,j}^n - u_{i,j}^n, u_{i,j}^{1,1} - u_{i,j}^{1,-1}), \\ \nabla^{0,-1} u_{i,j}^n &= \frac{1}{h}(u_{i,j}^{1,-1} - u_{i,j}^{-1,-1}, u_{i,j-1}^n - u_{i,j}^n), \\ \nabla^{-1,0} u_{i,j}^n &= \frac{1}{h}(u_{i-1,j}^n - u_{i,j}^n, u_{i,j}^{-1,1} - u_{i,j}^{-1,-1}), \\ \nabla^{0,1} u_{i,j}^n &= \frac{1}{h}(u_{i,j}^{1,1} - u_{i,j}^{-1,1}, u_{i,j+1}^n - u_{i,j}^n), \end{aligned} \quad (2.24)$$

Now we can define

$$\begin{aligned} Q_{ij}^{lm;n} &= \sqrt{\varepsilon^2 + |\nabla^{lm} u_{ij}^n|^2} \\ \bar{Q}_{ij}^{lm;n} &= \sqrt{\varepsilon^2 + \frac{1}{4} \sum_{|l|+|m|=1} |\nabla^{lm} u_{ij}^n|^2}, \end{aligned} \quad (2.25)$$

where $l, m \in \{-1, 0, 1\}$, $|l| + |m| = 1$, in the set of grid neighbors $N_{i,j}$. Therefore, the final discretized form of Equation (2.20) is given by [39]

$$u_{ij}^{n+1} - u_{ij}^n = \frac{\tau_S}{h^2} \bar{Q}_{ij}^{lm;n} \sum_{|l|+|m|=1} g_{ij}^{lm,\sigma} \frac{u_{i+l,j+m}^{n+1} - u_{ij}^{n+1}}{Q_{ij}^{lm;n}}, \quad (2.26)$$

where h^2 is the pixel area and $g_{ij}^{lm,\sigma} = g(|\nabla^{lm} I_{ij}^0|)$.

2.4 Experiments with 2D time-lapse data of macrophages

In the dataset we experiment with, macrophages are imaged from 30 minutes post-amputation to 6 hours post-amputation (0.5 – 6 *hpa*) with the imaging time step of 2 minutes, z step of 1 μm , and the pixel size of 0.347 μm . Then, the 3D+time images are projected into the 2D+time images, where the three-dimensional (3D) microscopy images are projected onto a plane with the maximum intensity of the 3D dataset in every pixel selected. Due to the image acquisition speed, the exposition time and fluorescence intensity are reduced, resulting in a low signal-to-noise ratio.

We first perform the histogram crop from the acquired images to ignore the noise effects from very high image intensity in a small pixel area. After the histogram crop, the image intensity is scaled to the interval $[0, 1]$ for applying space-time filtering. Then, the images obtained from space-time filtering are rescaled to the interval $[0, 255]$ to simply perform the local Otsu's method since histograms of images are usually described by the discrete distribution in a finite interval. To apply the SUBSURF method, two types of images are used; one is the original images after the histogram crop with the interval $[0, 1]$, and the other is the output of the local Otsu's method.

For solving space-time filtering and the SUBSURF method, the successive over-relaxation (SOR) method [40] is used. In our simulations, the relaxation factor of the SOR method was set to 1.8. The calculation was stopped when $\sum_{i=1}^M \sum_{j=1}^N |u_{i,j,k}^{n+1} - u_{i,j,k}^n| < 0.001$ for every k for space-time filtering and $\sum_{i=1}^M \sum_{j=1}^N |u_{i,j}^{n+1} - u_{i,j}^n| < 0.01$ for the SUBSURF method. The parameters we used are as follows: $\tau_F = 0.25$, $K = 100$, $\sigma = 0.1$ for space-time filtering, $s = 50$, $\delta = 0.5$ for the local Otsu's method, and $\tau_S = 0.25$, $K = 10$, $\sigma = 1$ for the SUBSURF method.

Figure 2.2 shows five different macrophages and the processing of the proposed segmentation. The local Otsu's method preceded by space-time filtering can capture the approximate shape of macrophages that have very weak image intensity. However, there can be some artifacts where pixels around the macrophages are also detected. As shown in the third column of the figure, the SUBSURF method can remove those artifacts where some detected noise around the objects and also make smoothen the inside and boundary of the macrophages. Figure 2.3 presents the results of the automatic segmentation for five macrophages, showing it works reliably for differently shaped macrophages no matter how complicated their boundaries are. In this figure, The fourth and fifth rows show the segmentation yields a few segmented regions for a single macrophage, in which four and two segmented regions, respectively. Therefore, it remains an open question of how to segment the whole bodies of macrophages with some areas of weak image intensity.

The performance of the presented segmentation method is evaluated quantitatively by using the mean Hausdorff distance of automatic and semi-automatic segmentation results; for the definition of the mean Hausdorff distance see equations (37)-(38) in [41]. The semi-automatic segmentation method, based on the Lagrangian approach [42], is used to create the "gold standard" for comparison, see also [12]. For the quantitative comparison, we choose one macrophage (the fifth macrophage in Figure 2.3). Also, the quality of the segmentation is measured by using the IoU (Jaccard) index [43] and the Sørensen–Dice coefficient [44, 45]. We see that the average of the mean Hausdorff distances for the macrophage is small compared to the size of macrophages. The

IoU index and Sørensen–Dice coefficient obtained from the proposed method indicate the results show reasonable performance. The perimeter, area, and circularity ($4\pi * \text{area}/\text{perimeter}^2$) are calculated for both automatic and semi-automatic segmentation in Figure 2.4(d)–(f), showing the geometrical information obtained from the proposed segmentation is close to the ones from the gold standard.

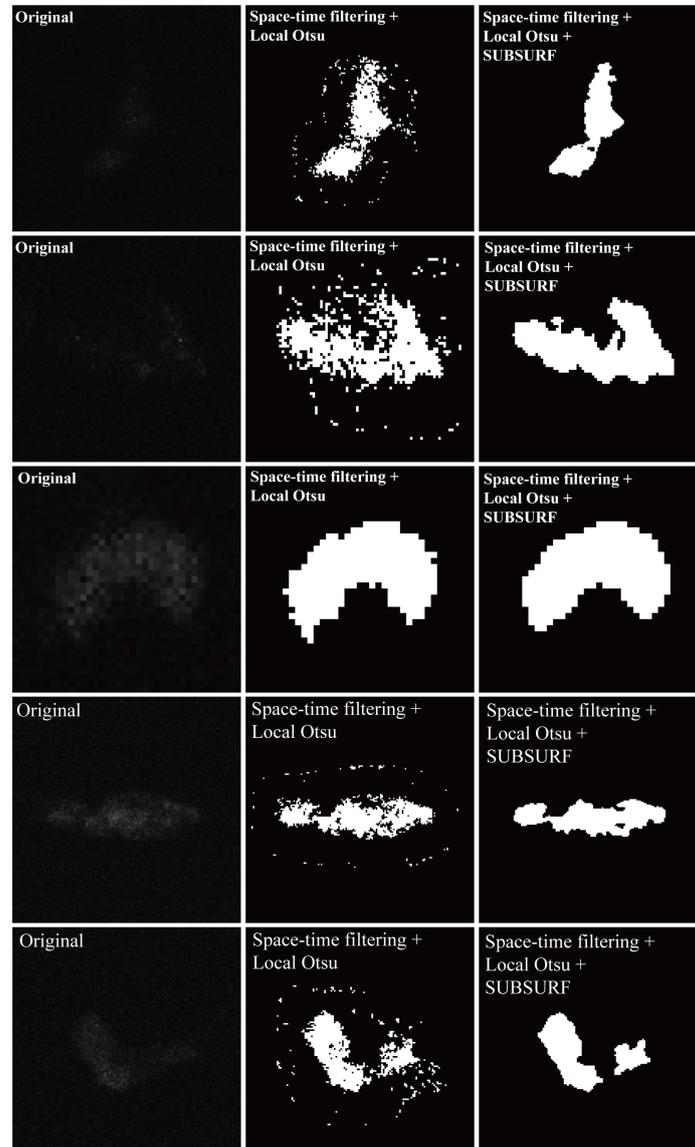


Figure 2.2: Five different macrophages and their processing by the proposed segmentation method. The first column shows the original images where the macrophages are hardly recognizable. The second column shows already recognizable macrophages in the images obtained by the local Otsu’s method preceded by the space-time filtering. The last column gives the results after the last segmentation step, the application of the SUBSURF method.

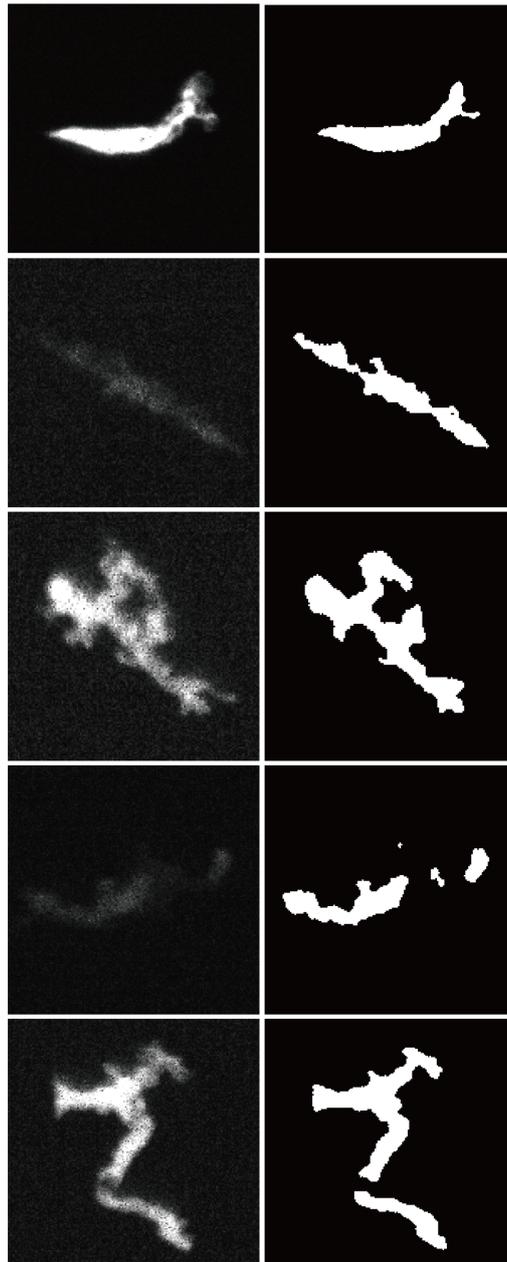


Figure 2.3: Five different macrophages from the original (left) and the segmented images using the combination of local Otsu's with the filtered images obtained from space-time filtering and the SUBSURF method (right).

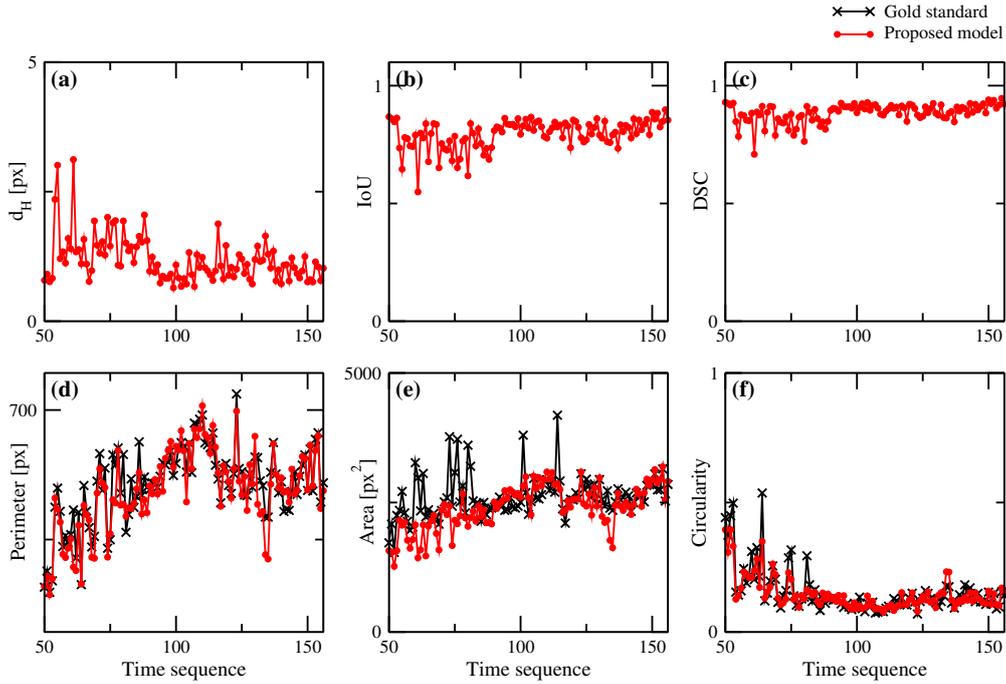


Figure 2.4: The quantitative comparison for a chosen macrophage. (a): The mean Hausdorff distance from the gold standard, (b): IoU (Jaccard) index, (c): Sørensen–Dice coefficient, (d): Perimeters of segmentations, (e): Areas of segmentations, and their (f): circularities.

3. Cell tracking

In this chapter, we propose a method for cell tracking based on segmented images. The approximated cell centers of all segmented cells are computed by solving the time-relaxed eikonal equation. Next, the approximate cell centers form trajectories when the segmented cells overlap each other in the temporal direction to trace the individual cell over time. Finally, these firstly formed trajectories are connected by computing a tangent allowing us to estimate the direction of movement of the cells. The two trajectories are connected by using the tangent calculation of a trajectory.

3.1 Detection of the approximate cell center

The time-relaxed eikonal equation is solved to find the cell centers in the segmentation results. As shown in Figure 2.2 and 2.3, the segmentation does not always extract the whole shape of some cells (macrophages). Therefore, the connected segmented subregions are used for obtaining the approximate centers of the cells, and we will call the connected segmented subregions segmented regions in short. We approximate the centers of the cells by finding the maxima of the distance function evaluated from the boundary of the segmented regions. The distance function will be obtained by solving the time-relaxed eikonal equation with the computation restricted only to the area of segmented regions. Since the centers—the local maxima of the distance function—are not exactly identical to the real cell centers, they will be called “approximate cell centers”. We describe the solution of the eikonal equation by the Rouy–Tourin scheme. The time-relaxed eikonal equation is written as

$$d_t + |\nabla d| = 1. \quad (3.1)$$

In every time slice $\theta \in [0, \theta_F]$, we solve equation(3.1) for the unknown function $d(\mathbf{x}, t, \theta)$ where $(\mathbf{x}, t) \in \Omega \times [0, T_E]$. The equation is discretized by the explicit scheme using the step size τ_D , and the Rouy–Tourin scheme is used for space discretization [46, 47, 48]. We solve Equation (3.1) in every 2D data slice. Let $d_{ij}^n(\theta)$ denote the approximate solution of Equation (3.1) at the time slice θ in pixel (i, j) at a discrete step $t^n = n\tau_D$. For

every (i, j) , the index set N_{ij} consists of all (l, m) such that $l, m \in \{-1, 0, 1\}$, $|l| + |m| = 1$, and then $D_{ij}^{lm}(\theta)$ is defined for any (l, m) as

$$D_{ij}^{lm}(\theta) = \left(\min \left(d_{i+l, j+m}^n(\theta) - d_{ij}^n(\theta), 0 \right) \right)^2. \quad (3.2)$$

In addition,

$$\begin{aligned} M_{ij}^{10}(\theta) &= \max \left(D_{ij}^{-1,0}(\theta), D_{ij}^{1,0}(\theta) \right), \\ M_{ij}^{01}(\theta) &= \max \left(D_{ij}^{0,-1}(\theta), D_{ij}^{0,1}(\theta) \right), \end{aligned} \quad (3.3)$$

are defined. Finally, the discretization of Equation (3.1) at time slice θ takes the following form,

$$d_{ij}^{n+1}(\theta) = d_{ij}^n(\theta) + \tau_D - \frac{\tau_D}{h} \sqrt{M_{ij}^{10}(\theta) + M_{ij}^{01}(\theta)}, \quad (3.4)$$

where $\tau_D = h/2$ is used for stability reasons. For the proposed cell tracking method, this equation is solved only inside the segmented regions according to the following process. The first step is to set $d_{ij}^0(\theta) = 0$ inside the segmented regions and $d_{ij}^0(\theta) = BIG$ outside the segmented regions; here, the value BIG is much greater than 0. Next, the numerical scheme in equation(3.4) is applied only inside the segmented regions. $d_{ij}^n(\theta)$ is fixed to 0 at the boundary of segmented regions by considering a pixel (i, j) is at the boundary in case that $d_{ij}^n(\theta) \neq BIG$ and there is at least one neighboring pixel which fulfills $d_{i+l, j+m}^n(\theta) = BIG$. The computation is stopped when the following condition is fulfilled $\sum_{\theta=0}^{\theta_F} \sum_{i=1}^M \sum_{j=1}^N |d_{ij}^{n+1}(\theta) - d_{ij}^n(\theta)| < \varepsilon_d$.

3.2 Extraction of partial trajectories

The obtained approximate centers of the segmented regions are connected when the segmented regions overlap in the temporal direction. The algorithm to connect the approximate cell centers in case of overlapping segmented regions using the backtracking approach is introduced in this Section. The algorithm yields the trajectories that connect the cells overlapping in the temporal direction—all these trajectories will be called partial trajectories.

Three sets of values will play a major role in the algorithm: $d_{ij}(\theta)$, $\mathcal{F}_{ij}(\theta)$, and $C^l(\theta)$, where (i, j) denote a pixel position, θ denotes a time slice and l denotes the cell center number.

First, the distance function value $d_{ij}(\theta)$ indicates whether a pixel is inside a segmented region or not. The pixel (i, j) at time slice θ is positioned inside the segmented region if $d_{ij}(\theta) \neq BIG$. Second, $\mathcal{F}_{ij}(\theta) = 1$ indicates the pixel (i, j) belongs to the segmented region which is already connected to another cell by a partial trajectory. Lastly, $C^l(\theta)$ represents the selected cell center, $l = 1, \dots, N^\theta$, where N^θ is the total number of segmented regions at the time step θ . With these definitions, the steps for linking the approximate cell centers are as follows:

1. For all pixels (i, j) and all time steps θ , $\mathcal{F}_{ij}(\theta)$ is set to 0 and $d_{ij}(\theta)$ is computed by the method equation(3.4).
2. Let θ_L be a time slice and let $\theta_L = \theta_F$ initially. The values of the distance function ($d_{ij}(\theta_L) \neq BIG$) inside every segmented region in time slice θ_L are inspected and the pixel having the maximal value of distance function inside the segmented region is found and designated as approximate cell center, $C^l(\theta_L) = (C_1^l(\theta_L), C_2^l(\theta_L))$, $l = 1, \dots, N^\theta$.
3. Let $\theta = \theta_L$. In a backtracking manner, we look for overlapping segmented regions by performing steps (a)-(b): for $l = 1, \dots, N^\theta$, $C^l(\theta)$ is projected onto the spatial plane of the previous time slice $\theta - 1$. Let denote the projected point as $P(C^l(\theta))$, where $P(C^l(\theta)) = (C_1^l(\theta), C_2^l(\theta), \theta - 1)$.

- (a) The case when an approximate cell center is projected inside some segmented region:
 If $d_{ij}(\theta - 1) \neq BIG$ for $(i, j) = (C_1^l(\theta), C_2^l(\theta))$, the approximate cell center at $\theta - 1$ is found by searching for the maximum value of the distance function inside the segmented region at time $\theta - 1$, and the approximate cell center is denoted by $C^l(\theta - 1) = (C_1^l(\theta - 1), C_2^l(\theta - 1))$. Also, $\mathcal{F}_{ij}(\theta - 1)$ is changed to 1 for all pixels (i, j) inside the corresponding segmented region. After finding the approximate cell center $C^l(\theta - 1)$, it is connected with $C^l(\theta)$, forming a section of the partial trajectory (see Figure 3.1).
- (b) The case when the projected cell center is not inside of any segmented region at time $\theta - 1$:
 If $d_{ij}(\theta - 1) = BIG$ for $(i, j) = (C_1^l(\theta), C_2^l(\theta))$, let $S^l(\theta)$ be a set of all pixels (i, j) belonging to the l^{th} segmented region at time θ . Then $d_{ij}(\theta - 1)$ is inspected for all pixels (i, j) in $S^l(\theta)$. The inspection is stopped if $d_{ij}(\theta - 1) \neq BIG$ for some $(i, j) = (p^*, q^*)$ and denoting such point $S_{p^*, q^*}^l(\theta)$, or if all pixels in $S^l(\theta)$ are inspected without finding such a point.
- i. If a point $S_{p^*, q^*}^l(\theta)$ exists, the approximate cell center $C^l(\theta - 1) = (C_1^l(\theta - 1), C_2^l(\theta - 1))$ is found like in the step 3(a) but starting from $P(S_{p^*, q^*}^l(\theta))$, and $\mathcal{F}_{ij}(\theta - 1)$ is set to 1 for all pixels inside the segmented region at time $\theta - 1$ to which $P(S_{p^*, q^*}^l(\theta))$ belongs to. After finding the approximate cell center, $C^l(\theta - 1)$ is connected with $C^l(\theta)$, forming a section of the partial trajectory.
 - ii. If a point $S_{p^*, q^*}^l(\theta)$ does not exist, the approximate cell center is not designated because there is no overlap of the segmented region l at time θ with any segmented region at time $\theta - 1$.
4. Step 3 is repeated by decreasing θ by one until $\theta = 1$.
5. θ_L is decreased by one and $d_{ij}(\theta_L)$ and $\mathcal{F}_{ij}(\theta_L)$ are checked for all (i, j) .
 If there is a pixel that fulfills $d_{ij}(\theta_L) \neq BIG$ and $\mathcal{F}_{ij}(\theta_L) = 0$, we consider that the pixel is inside a segmented region at θ_L non-overlapping with segmented regions at $\theta_L + 1$. The approximate cell center of the region is found like in step 2, and then steps 3 and 4 are repeated.
6. Step 5 is repeated until $\theta_L = 1$.

Let us note that in steps 3(a) and 3(b), if there are several maxima of the distance function in the inspected segmented region, then the cell center is chosen as the first one found. In step 3, the trajectories can remain disconnected if there is no overlap of cells and condition 3(b)-ii is fulfilled. In the top panel of Figure 3.3, such partial trajectories are depicted inside the 3D spatial-temporal structure formed by stacking segmented regions in the temporal direction. This shows that the algorithm works correctly for the overlapped cells, and the extracted partial trajectories appear as expected.

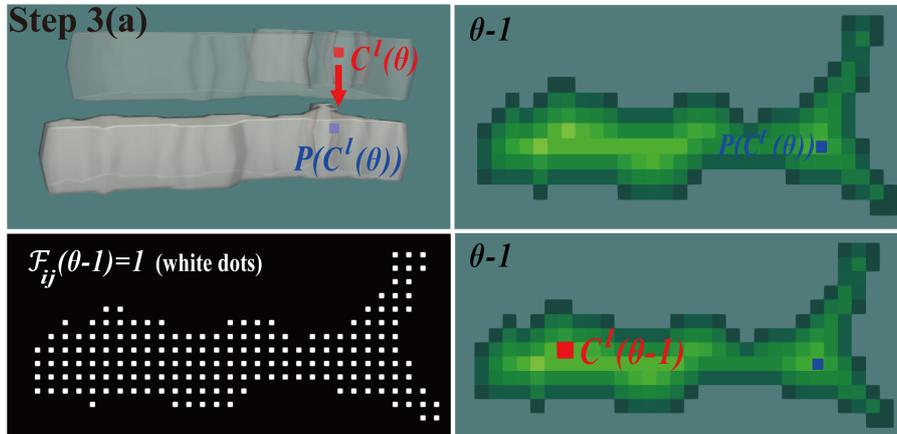


Figure 3.1: Schematic picture of step 3(a) of the proposed algorithm. The cells in the top-left panel are amplified along the time axis for better visualization. The blue dot denotes the projected coordinate.

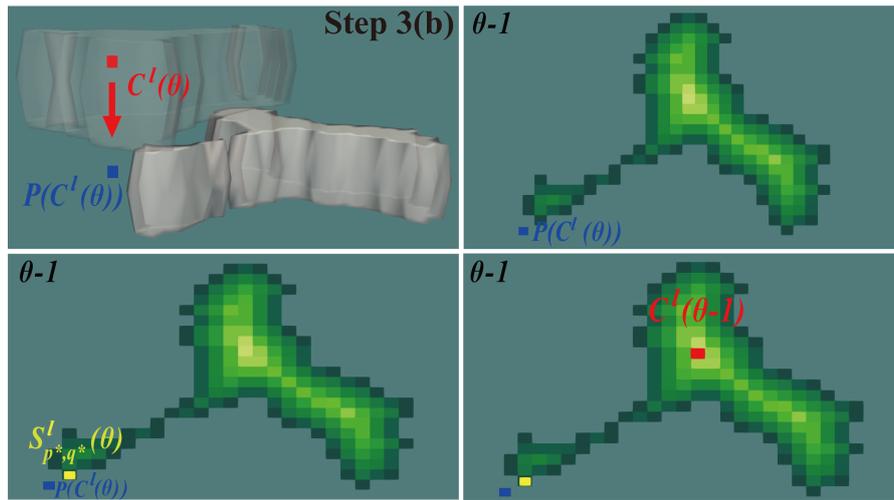


Figure 3.2: Schematic picture of steps 3(b) of the proposed algorithm. The cells in the top-left panel are amplified along the time axis for better visualization. The yellow dot denotes the inspected coordinate where $d_{ij}(\theta - 1) \neq BIG$. Step 3(b)-i is shown in the bottom panels.

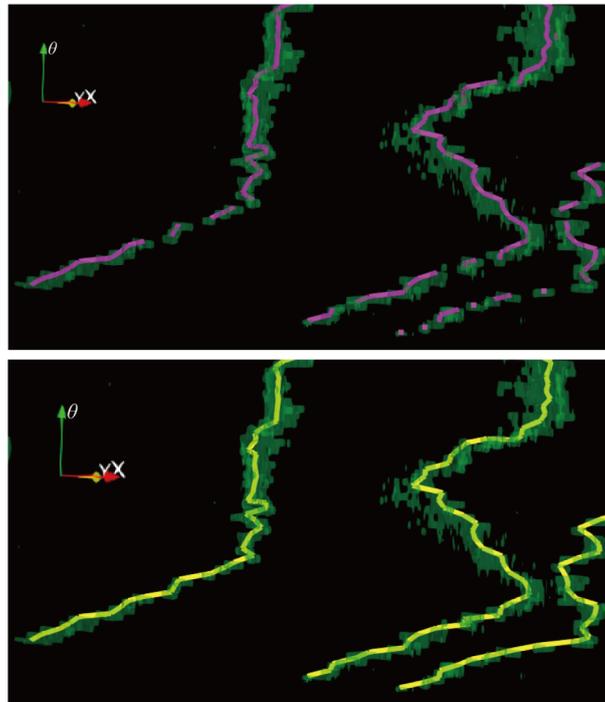


Figure 3.3: Top: Partial trajectories of several macrophages. Bottom: Connected trajectories from the partial trajectories in the top panel. The time axis is amplified for better visualization.

3.3 Estimation of cell movement direction

The tangent approximation by the backward finite difference is used to estimate the position of a point in the next time step of a partial trajectory. Similarly, the forward finite difference for the tangent calculation is used for the estimation of the point at the previous time step of the partial trajectory. In the tangent calculation, third-order accuracy is maximally considered, and thus there are three forms of tangent approximation depending on the number of points in the partial trajectory. The tangents computed with the third order accuracy using the backward and forward finite difference approximations are given by [49]

$$\begin{aligned}
V^b(\mathbf{r}_\theta) &= \frac{1}{\Delta\theta} \left(\frac{11}{6}\mathbf{r}_\theta - 3\mathbf{r}_{\theta-1} + \frac{3}{2}\mathbf{r}_{\theta-2} - \frac{1}{3}\mathbf{r}_{\theta-3} \right), \\
V^f(\mathbf{r}_\theta) &= \frac{1}{\Delta\theta} \left(-\frac{11}{6}\mathbf{r}_\theta + 3\mathbf{r}_{\theta+1} - \frac{3}{2}\mathbf{r}_{\theta+2} + \frac{1}{3}\mathbf{r}_{\theta+3} \right),
\end{aligned} \tag{3.5}$$

where $\Delta\theta$ means the size of the time slice difference, and $\mathbf{r}_\theta = (x_\theta, y_\theta)$ is the point of the partial trajectory in the time slice θ . Similarly, second and first-order accuracy approximations for backward and forward finite differences are given by

$$\begin{aligned}
V^b(\mathbf{r}_\theta) &= \frac{1}{\Delta\theta} \left(\frac{3}{2}\mathbf{r}_\theta - 2\mathbf{r}_{\theta-1} + \frac{1}{2}\mathbf{r}_{\theta-2} \right), \\
V^f(\mathbf{r}_\theta) &= \frac{1}{\Delta\theta} \left(-\frac{3}{2}\mathbf{r}_\theta + 2\mathbf{r}_{\theta+1} - \frac{1}{2}\mathbf{r}_{\theta+2} \right),
\end{aligned} \tag{3.6}$$

and

$$\begin{aligned}
V^b(\mathbf{r}_\theta) &= \frac{1}{\Delta\theta} (\mathbf{r}_\theta - \mathbf{r}_{\theta-1}), \\
V^f(\mathbf{r}_\theta) &= \frac{1}{\Delta\theta} (-\mathbf{r}_\theta + \mathbf{r}_{\theta+1}).
\end{aligned} \tag{3.7}$$

Let us consider a partial trajectory with a time step range $[a\Delta\theta, b\Delta\theta]$, a, b integers, and denote the positions of the cell center at $a\Delta\theta$ and $b\Delta\theta$ by \mathbf{r}_a and \mathbf{r}_b , respectively. Then, the position of the cell center at $(a-1)\Delta\theta$ can be estimated from the tangent at time step $a\Delta\theta$. For example, if the partial trajectory contains more than three points, the tangent obtained by using the forward finite difference at $a\Delta\theta$ is given by

$$V^f(\mathbf{r}_a) = \frac{1}{\Delta\theta} \left(-\frac{11}{6}\mathbf{r}_a + 3\mathbf{r}_{a+1} - \frac{3}{2}\mathbf{r}_{a+2} + \frac{1}{3}\mathbf{r}_{a+3} \right) \tag{3.8}$$

and the tangent at $(a-1)\Delta\theta$ would be

$$V^f(\mathbf{r}_{a-1}) = \frac{1}{\Delta\theta} \left(-\frac{11}{6}\mathbf{r}_{a-1} + 3\mathbf{r}_a - \frac{3}{2}\mathbf{r}_{a+1} + \frac{1}{3}\mathbf{r}_{a+2} \right). \tag{3.9}$$

Assuming $T^f(\mathbf{r}_{a-1}) = T^f(\mathbf{r}_a)$, i.e. the uniform directional motion of non-overlapping macrophages, we see that \mathbf{r}_{a-1} is the only unknown in the equation and can be easily obtained. Similarly, the tangent at time steps $b\Delta\theta$ yields the estimated cell center at time step $(b+1)\Delta\theta$ using the backward finite difference. So, the estimated points on trajectories in the time slice without overlap of cells are given by

$$\begin{aligned}
\mathbf{r}_{a-1} &= -\frac{6}{11}V^f(\mathbf{r}_a) \cdot \Delta\theta + \frac{18}{11}\mathbf{r}_a - \frac{9}{11}\mathbf{r}_{a+1} + \frac{2}{11}\mathbf{r}_{a+2}, \quad b-a > 2, \\
\mathbf{r}_{a-1} &= -\frac{2}{3}V^f(\mathbf{r}_a) \cdot \Delta\theta + \frac{4}{3}\mathbf{r}_a - \frac{1}{3}\mathbf{r}_{a+1}, \quad b-a = 2,
\end{aligned} \tag{3.10}$$

$$\mathbf{r}_{a-1} = -V^f(\mathbf{r}_a) \cdot \Delta\theta + \mathbf{r}_a, \quad b-a = 1,$$

and

$$\begin{aligned}\mathbf{r}_{b+1} &= \frac{6}{11}V^b(\mathbf{r}_b) \cdot \Delta\theta + \frac{18}{11}\mathbf{r}_b - \frac{9}{11}\mathbf{r}_{b-1} + \frac{2}{11}\mathbf{r}_{b-2} \quad , b-a > 2, \\ \mathbf{r}_{b+1} &= \frac{2}{3}V^b(\mathbf{r}_b) \cdot \Delta\theta + \frac{4}{3}\mathbf{r}_b - \frac{1}{3}\mathbf{r}_{b-1} \quad , b-a = 2, \\ \mathbf{r}_{b+1} &= V^b(\mathbf{r}_b) \cdot \Delta\theta + \mathbf{r}_b \quad , b-a = 1.\end{aligned}\tag{3.11}$$

3.4 Connection of partial trajectories

The connection of partial trajectories is carried out when the estimated center of the cell \mathbf{r}_{es} given by \mathbf{r}_{a-1} or \mathbf{r}_{b+1} in equations (3.10) or (3.11), is positioned near the end point \mathbf{r}_e of some existing partial trajectory ending at time slice $(a-1)\Delta\theta$ or starting at $(b+1)\Delta\theta$. It means, we check the condition

$$|\mathbf{r}_{es} - \mathbf{r}_e| \leq \Delta r \tag{3.12}$$

where Δr is a parameter, and if it is fulfilled then the partial trajectories are connected.

Figure 3.4(a) shows two partial trajectories denoted by α and β . The red dot in the figure represents the estimated cell center $\mathbf{r}_{es;\alpha}$ computed from the α trajectory with the backward finite difference approximation. The α and β trajectories are connected if the beginning point of the β trajectory and the estimated cell center from α trajectory is located within the neighborhood Δr ; Figure 3.4(b) shows the connected trajectory in such case. The condition (3.12) is written for the case when the difference of time slices between endpoints of partial trajectories is equal to 1. However, the partial trajectories are connected similarly when the difference of time slices is equal to 2, i.e. if two estimated points obtained from two partial trajectories (one in a forward manner and one in a backward manner) are located in the same time slice and within the Δr neighborhood. The connection of partial trajectories by using the above approach is shown in the bottom panel of Figure 3.3.

Furthermore, the tangent calculation is also used to connect the partial trajectories if their last points are located close to each other in several time slices. It can happen if the segmentation of a single macrophage contains several fractions in a few time slices. Figure 3.4(c) shows two such partial trajectories γ and λ . As shown in the blue circle, there are three common time slices where γ and λ have trajectory points close to each other. To connect those kinds of partial trajectories, we again calculate the estimated point of the partial trajectory using the tangent approximation and check if there is a point \mathbf{r}_j of another trajectory in a close neighborhood of the estimated point. If yes, then also a difference between the time slice of \mathbf{r}_j and the time slice of the endpoint \mathbf{r}_e of its trajectory is checked. In other words, we check the number of common time slices Θ_c where two close trajectories appear simultaneously. For instance, in the case of Figure 3.4(c), $\Theta_c = 3$. Finally, two trajectories are connected if the following two conditions are fulfilled: $|\mathbf{r}_{es} - \mathbf{r}_j| \leq \Delta r_2$ and $\Theta_c \leq \Delta r_\theta$. In Figure 3.4(d) we plot by the red line such connected trajectory.

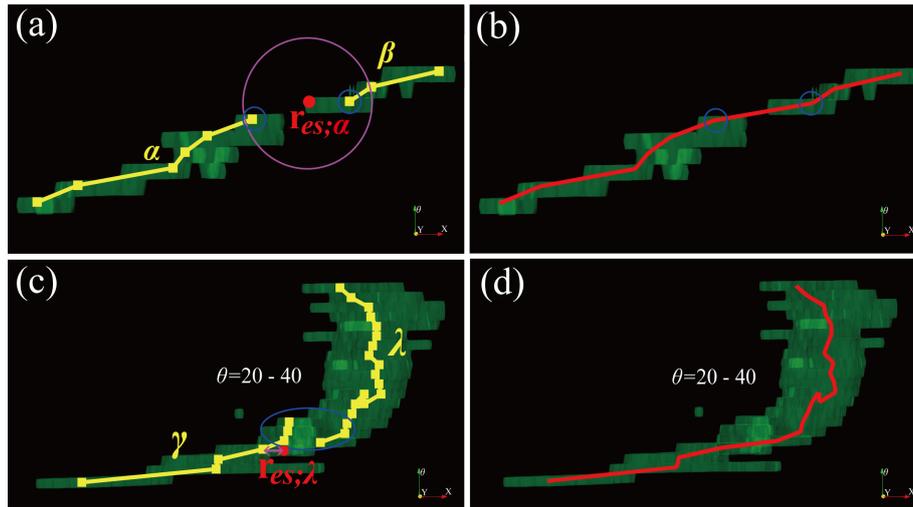


Figure 3.4: Two different partial trajectories α and β (a), and the connected trajectory containing α and β (b). The blue circles show the last point of the partial trajectory α and the beginning point of β . Two different trajectories γ and λ (c), and the connected trajectory containing γ and λ (d). The blue circle shows the points of the two trajectories in the common time slices.

3.5 Experiments with 2D time-lapse data of macrophages

The segmented images obtained from the segmentation in the previous chapter are used for experimenting with the proposed tracking method. For this dataset, the condition (3.12) with $\Delta r = 60$ was first used to connect the partial trajectories and then parameters $\Delta r_2 = 120$ and $\Delta r_\theta = 5$ were used to avoid closed trajectories due to macrophage segmentation split. We note that Δr_2 can be chosen less sensitively than Δr since many trajectories are already connected.

The final trajectories in the whole spatial domain are visualized in Figure 3.5. There are 9 detected macrophages at the beginning and 12 macrophages are shown at the last time slice as new macrophages appear and disappear over time. The site of the wound is positioned on the right side and many macrophages migrate toward the wound. Especially, the macrophage denoted by violet located at the top of the pictures from $\theta = 115$ to $\theta = 156$ shows very fast movement which implies the macrophage yields many partial trajectories, and our method enables us to connect them.

In the second row of the figure, the two trajectories (purple and brown) inside the blue rectangle are located in the same cell, but they are shown differently due to the problem that the segmentation cannot extract the entire shapes of macrophages, and their number of the common time slices Θ_c is greater than 6.

The number of trajectories before and after the first connection by using condition (3.12) (see also Figure 3.4(a)–(b)) was 930 and 234, respectively. After the second connection of closed common trajectories (see Figure 3.4(c)–(d)), the number of extracted trajectories decreased again significantly to 69. In addition, the average length of extracted trajectories after the second connection increased from 69.53 to 363.29 in pixel units.

The mean Hausdorff distance computed for three selected trajectories, see panels (a)–(c) of Figure 3.6, was 3.13, 4, 35, and 2.40 in units of pixels, which is very low in comparison to the average length of trajectories and it shows again the high accuracy of tracking for this dataset. The comparison of the average distance computed by averaging the Euclidean distance between points at each time frame of manual and the proposed tracking is presented in Table 3.1. This distance is bigger than the mean Hausdorff distance but still small compared to the average size of macrophages.

Further check of automatic tracking accuracy we performed by counting the number of correct and wrong links in every time slice by visual inspection. We define the time slice tracking accuracy as the ratio between the number of correct links and the total number of links detected visually in one time slice in a forward manner, and we define the mean accuracy of tracking as the average of the tracking accuracy over all time slices. As a result, the mean accuracy of tracking was 0.974.

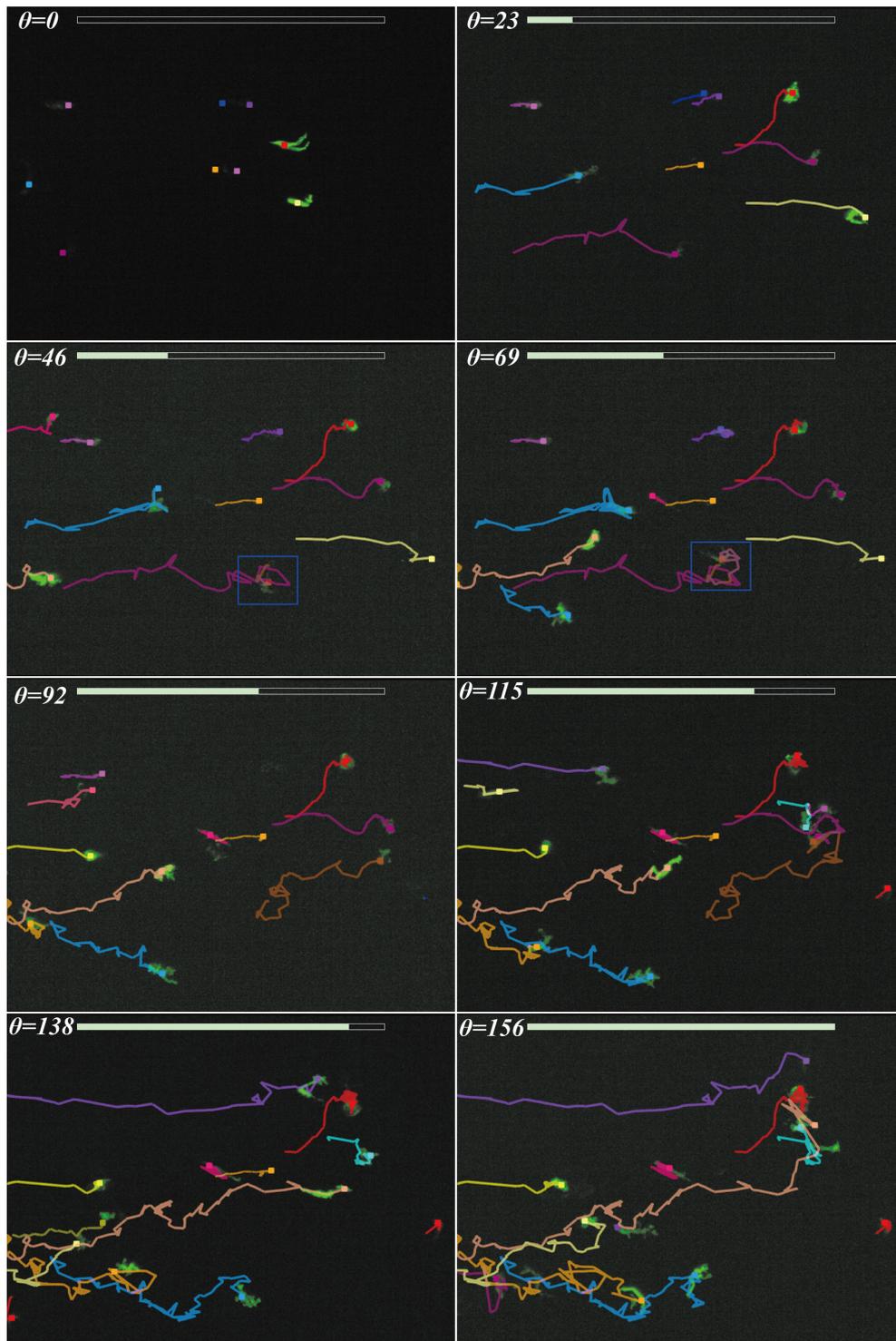


Figure 3.5: Final trajectories in eight subsequent time slices. The final time slice is $\theta_F = 156$. Here, the size of the whole spatial domain is $1758 * 1306$ pixels.

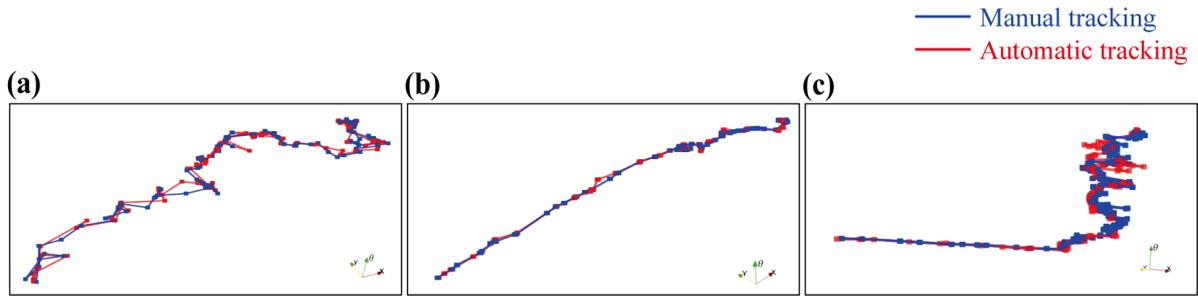


Figure 3.6: Extracted trajectories of three different macrophages from manual and automatic tracking methods. The numbers of points of (a)–(c) are 57, 109, and 157, respectively. The blue lines show manual tracking in Fiji [50], and the red lines show our proposed tracking method.

	# of points	d_H [px]	d_{avg} [px]	L_{manual} [px]	L_{auto} [px]
(a)	109	3.13	17.33	2425.88	3026.21
(b)	57	4.35	19.78	1887.99	1920.75
(c)	157	2.40	15.71	2039.42	1720.62

Table 3.1: The two different types of the distance between trajectories obtained from manual and proposed tracking. Here, d_H and d_{avg} denote the mean Hausdorff distance between two curves and the average of the Euclidean distance between two points at each time slice. L_{manual} is the total length of trajectories by manual tracking, and L_{auto} is the total length obtained from proposed tracking.

List of publications

Park, Seol Ah – Sipka, Tamara – Kriva, Zuzana – Ambroz, Martin – Kollár, Michal – Kósa, Balázs – Nguyen-Chi, Mai – Lutfalla, Georges – Mikula, Karol. Macrophage image segmentation by Thresholding and subjective surface Method, *Tatra Mountains Mathematical Publications*, **75(1)**, 103–120 (2020). <https://doi.org/10.2478/tmmp-2020-0007>, IF:0.47.

Uba, Markjoe Olunna – Mikula, Karol – Park, Seol Ah. 4D Segmentation Algorithm with application to 3D+ time Image Segmentation, *Japan Journal of Industrial and Applied Mathematics*, **40**, 109–139 (2023). <https://doi.org/10.1007/s13160-022-00519-w>, IF: 0.682.

Nguyen-Chi, Mai – Sipka, Tamara – Park, Seol Ah – Ozbilgic, Resul – Balas, Laurence – Durand, Thierry – Mikula, Karol – Lutfalla, Georges. Macrophages undergo a behavioural switch during wound healing in zebrafish, *Free Radical Biology and Medicine*, **192**, 200-212 (2022). <https://doi.org/10.1016/j.freeradbiomed.2022.09.021>, IF: 8.101.

Park, Seol Ah – Sipka, Tamara – Kriva, Zuzana – Lutfalla, Georges – Nguyen-Chi, Mai – Mikula, Karol. Segmentation-based tracking of macrophages in 2D+time microscopy movies inside a living animal, *Computers in Biology and Medicine*, **153**, 106499, (2023). <https://doi.org/10.1016/j.compbiomed.2022.106499>, IF: 6.698.

References

- [1] Neda Emami, Zahra Sedaei, and Reza Ferdousi. Computerized cell tracking: current methods, tools and challenges. *Visual Informatics*, 5(1):1–13, 2021.
- [2] Chentao Wen, Takuya Miura, Venkatakaushik Voleti, Kazushi Yamaguchi, Motosuke Tsutsumi, Kei Yamamoto, Kohei Otomo, Yukako Fujie, Takayuki Teramoto, Takeshi Ishihara, et al. 3deecelltracker, a deep learning-based pipeline for segmenting and tracking cells in 3d time lapse images. *Elife*, 10:e59187, 2021.
- [3] Marcus Wagner, René Hänsel, Sarah Reinke, Julia Richter, Michael Altenbuchinger, Ulf-Dietrich Braumann, Rainer Spang, Markus Löffler, and Wolfram Klapper. Automated macrophage counting in dlbc1 tissue samples: a rof filter based approach. *Biological procedures online*, 21(1):13, 2019.
- [4] Hassan M Rostam, Paul M Reynolds, Morgan R Alexander, Nikolaj Gadegaard, and Amir M Ghaemmaghami. Image based machine learning for identification of macrophage subsets. *Scientific reports*, 7(1):1–11, 2017.
- [5] José Alonso Solís-Lemus, Brian Stramer, Greg Slabaugh, and Constantino Carlos Reyes-Aldasoro. Segmentation and shape analysis of macrophages using anglegram analysis. *Journal of Imaging*, 4(1):2, 2018.
- [6] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015.
- [7] Uwe Schmidt, Martin Weigert, Coleman Broaddus, and Gene Myers. Cell detection with star-convex polygons. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 265–273. Springer, 2018.
- [8] Thorsten Falk, Dominic Mai, Robert Bensch, Özgün Çiçek, Ahmed Abdulkadir, Yassine Marrakchi, Anton Böhm, Jan Deubner, Zoe Jäckel, Katharina Seiwald, et al. U-net: deep learning for cell counting, detection, and morphometry. *Nature methods*, 16(1):67–70, 2019.
- [9] Jean-Baptiste Lugagne, Haonan Lin, and Mary J Dunlop. Delta: Automated cell segmentation, tracking, and lineage reconstruction using deep learning. *PLoS computational biology*, 16(4):e1007673, 2020.
- [10] Carsen Stringer, Tim Wang, Michalis Michaelos, and Marius Pachitariu. Cellpose: a generalist algorithm for cellular segmentation. *Nature methods*, 18(1):100–106, 2021.
- [11] Soham Mandal and Virginie Uhlmann. Splinedist: Automated cell segmentation with spline curves. In *2021 IEEE 18th International Symposium on Biomedical Imaging (ISBI)*, pages 1082–1086. IEEE, 2021.
- [12] Seol Ah Park, Tamara Sipka, Zuzana Kriva, Martin Ambroz, Michal Kollár, Balázs Kósa, Mai Nguyen-Chi, Georges Lutfalla, and Karol Mikula. Macrophage image segmentation by thresholding and subjective surface method. *Tatra Mountains Mathematical Publications*, 75(1):103–120, 2020.
- [13] Alessandro Sarti, Karol Mikula, and Fiorella Sgallari. Nonlinear multiscale analysis of three-dimensional echocardiographic sequences. *IEEE Transactions on Medical Imaging*, 18(6):453–466, 1999.

- [14] Alessandro Sarti, Ravi Malladi, and James A Sethian. Subjective surfaces: A method for completing missing boundaries. *Proceedings of the National Academy of Sciences*, 97(12):6258–6263, 2000.
- [15] Khuloud Jaqaman, Dinah Loerke, Marcel Mettlen, Hirotaka Kuwata, Sergio Grinstein, Sandra L Schmid, and Gaudenz Danuser. Robust single-particle tracking in live-cell time-lapse sequences. *Nature methods*, 5(8):695–702, 2008.
- [16] Visakan Kadirkamanathan, Sean R Anderson, Stephen A Billings, Xiliang Zhang, Geoffrey R Holmes, Constantino C Reyes-Aldasoro, Philip M Elks, and Stephen A Renshaw. The neutrophil’s eye-view: Inference and visualisation of the chemoattractant field driving cell chemotaxis in vivo. *PloS one*, 7(4):e35182, 2012.
- [17] Fernando Amat, William Lemon, Daniel P Mossing, Katie McDole, Yinan Wan, Kristin Branson, Eugene W Myers, and Philipp J Keller. Fast, accurate reconstruction of cell lineages from large-scale fluorescence microscopy data. *Nature methods*, 11(9):951–958, 2014.
- [18] Karol Mikula, Róbert Špir, Michal Smíšek, Emmanuel Faure, and Nadine Peyri eras. Nonlinear pde based numerical methods for cell tracking in zebrafish embryogenesis. *Applied Numerical Mathematics*, 95:250–266, 2015.
- [19] Emmanuel Faure, Thierry Savy, Barbara Rizzi, Camilo Melani, Olga Stařova, Dimitri Fabr eges, Róbert Špir, Mark Hammons, Róbert  underlık, Ga elle Recher, et al. A workflow to process 3d+ time microscopy images of developing organisms and reconstruct their cell lineage. *Nature communications*, 7(1):1–10, 2016.
- [20] Robert Spir, Karol Mikula, and Nadine Peyrieras. Cell lineage tree reconstruction from time series of 3d images of zebrafish embryogenesis. In *Asian Conference on Computer Vision*, pages 539–554. Springer, 2016.
- [21] Jean-Yves Tinevez, Nick Perry, Johannes Schindelin, Genevieve M Hoopes, Gregory D Reynolds, Emmanuel Laplantine, Sebastian Y Bednarek, Spencer L Shorte, and Kevin W Eliceiri. Trackmate: An open and extensible platform for single-particle tracking. *Methods*, 115:80–90, 2017.
- [22] Hideya Aragaki, Katsunori Ogoh, Yohei Kondo, and Kazuhiro Aoki. Lim tracker: a software package for cell tracking and analysis with advanced interactivity. *Scientific reports*, 12(1):1–14, 2022.
- [23] CC Reyes-Aldasoro, S Akerman, and GM Tozer. Measuring the velocity of fluorescently labelled red blood cells with a keyhole tracking algorithm. *Journal of microscopy*, 229(1):162–173, 2008.
- [24] Constantino Carlos Reyes-Aldasoro, Yifan Zhao, Daniel Coca, Stephen A Billings, Visakan Kadirkamanathan, Gillian M Tozer, and Stephen A Renshaw. Analysis of immune cell function using in vivo cell shape analysis and tracking. 2009.
- [25] Katherine M Henry, Luke Pase, Carlos Fernando Ramos-Lopez, Graham J Lieschke, Stephen A Renshaw, and Constantino Carlos Reyes-Aldasoro. Phagosight: an open-source matlab® package for the analysis of fluorescent neutrophil and macrophage migration in a zebrafish model. *PloS one*, 8(8):e72636, 2013.
- [26] Sam Cooper, Alexis R Barr, Robert Glen, and Chris Bakal. NucliTrack: an integrated nuclei tracking application. *Bioinformatics*, 33(20):3320–3322, 2017.
- [27] Vladimır Ulman, Martin Mařka, Klas EG Magnusson, Olaf Ronneberger, Carsten Haubold, Nathalie Harder, Pavel Matula, Petr Matula, David Svoboda, Miroslav Radojevic, et al. An objective comparison of cell-tracking algorithms. *Nature Methods*, 14(12):1141–1152, 2017.
- [28] Nisha Ramesh and Tolga Tasdizen. Semi-supervised learning for cell tracking in microscopy images. In *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, pages 948–951. IEEE, 2018.

- [29] David E Hernandez, Steven W Chen, Elizabeth E Hunter, Edward B Steager, and Vijay Kumar. Cell tracking with deep learning and the viterbi algorithm. In *2018 International Conference on Manipulation, Automation and Robotics at Small Scales (MARSS)*, pages 1–6. IEEE, 2018.
- [30] Hsieh-Fu Tsai, Joanna Gajda, Tyler FW Sloan, Andrei Rares, and Amy Q Shen. Usiigaci: Instance-aware cell tracking in stain-free phase contrast microscopy enabled by machine learning. *SoftwareX*, 9:230–237, 2019.
- [31] Marloes Arts, Ihor Smal, Maarten W Paul, Claire Wyman, and Erik Meijering. Particle mobility analysis using deep learning and the moment scaling spectrum. *Scientific Reports*, 9(1):1–10, 2019.
- [32] Junjie Wang, Xiaohong Su, Lingling Zhao, and Jun Zhang. Deep reinforcement learning for data association in cell tracking. *Frontiers in Bioengineering and Biotechnology*, 8:298, 2020.
- [33] Rafael Iriya, Wenwen Jing, Karan Syal, Manni Mo, Chao Chen, Hui Yu, Shelley E Haydel, Shaopeng Wang, and Nongjian Tao. Rapid antibiotic susceptibility testing based on bacterial motion patterns with long short-term memory neural networks. *IEEE Sensors Journal*, 20(9):4940–4950, 2020.
- [34] Samira Masoudi, Afsaneh Razi, Cameron HG Wright, Jesse C Gatlin, and Ulas Bagci. Instance-level microtubule tracking. *IEEE Transactions on Medical Imaging*, 39(6):2061–2075, 2020.
- [35] Zhichao Liu, Luhong Jin, Jincheng Chen, Qiuyu Fang, Sergey Ablameyko, Zhaozheng Yin, and Yingke Xu. A survey on applications of deep learning in microscopy image analysis. *Computers in Biology and Medicine*, 134:104523, 2021.
- [36] Luis Alvarez, Frédéric Guichard, Pierre-Louis Lions, and Jean-Michel Morel. Axioms and fundamental equations of image processing. *Archive for rational mechanics and analysis*, 123(3):199–257, 1993.
- [37] Karol Mikula and Mariana Remešíková. Finite volume schemes for the generalized subjective surface equation in image segmentation. *Kybernetika*, 45(4):646–656, 2009.
- [38] Lawrence C Evans and Joel Spruck. Motion of level sets by mean curvature. ii. *Transactions of the american mathematical society*, 330(1):321–332, 1992.
- [39] Karol Mikula, N Peyriéras, M Remešíková, and A Sarti. 3d embryogenesis image segmentation by the generalized subjective surface method using the finite volume technique. *Finite Volumes for Complex Applications V: Problems and Perspectives*, pages 585–592, 2008.
- [40] Richard Barrett, Michael Berry, Tony F Chan, James Demmel, June Donato, Jack Dongarra, Victor Eijkhout, Roldan Pozo, Charles Romine, and Henk Van der Vorst. *Templates for the solution of linear systems: building blocks for iterative methods*. SIAM, 1994.
- [41] Karol Mikula, Jozef Urbán, Michal Kollár, Martin Ambroz, Ivan Jarolímek, Jozef Šibík, and Mária Šibíková. An automated segmentation of natura 2000 habitats from sentinel-2 optical data. *Discrete & Continuous Dynamical Systems-S*, 14(3):1017, 2021.
- [42] Karol Mikula, Jozef Urbán, Michal Kollár, Martin Ambroz, Ivan Jarolímek, Jozef Šibík, and Mária Šibíková. Semi-automatic segmentation of natura 2000 habitats in sentinel-2 satellite images by evolving open curves. *Discrete & Continuous Dynamical Systems-S*, page 0, 2018.
- [43] Paul Jaccard. The distribution of the flora in the alpine zone. 1. *New Phytologist*, 11(2):37–50, 1912.
- [44] Lee R Dice. Measures of the amount of ecologic association between species. *Ecology*, 26(3):297–302, 1945.
- [45] Thorvald A Sorensen. A method of establishing groups of equal amplitude in plant sociology based on similarity of species content and its application to analyses of the vegetation on danish commons. *Biol. Skar.*, 5:1–34, 1948.