LEARNING TO DETECT AND TRACK CELLS FOR QUANTITATIVE ANALYSIS OF TIME-LAPSE MICROSCOPIC IMAGE SEQUENCES

Pedro D. Kostelec**, Leo M. Carlin*†*, Ben Glocker**

* Biomedical Image Analysis Group, Department of Computing
† Leukocyte Biology Section, National Heart & Lung Institute
Imperial College London, UK

ABSTRACT

Studying the behaviour of cells using time-lapse microscopic imaging requires automated processing pipelines that enable quantitative analysis of a large number of cells. We propose a pipeline based on state-of-the-art methods for background motion compensation, cell detection, and tracking which are integrated into a novel semi-automated, learning based analysis tool. Motion compensation is performed by employing an efficient nonlinear registration method based on powerful discrete graph optimisation. Robust detection and tracking of cells is based on classifier learning which only requires a small number of manual annotations. Cell motion trajectories are generated using a recent global data association method and linear programming. Our approach is robust to the presence of significant motion and imaging artifacts. Promising results are presented on different sets of *in-vivo* fluorescent microscopic image sequences.

Index Terms— Motion Compensation, Cell Detection and Tracking, Fluorescent Time-lapse Microscopic Imaging

1. INTRODUCTION

To contribute to homeostasis, protective immunity and pathology, leukocytes (white blood cells) have to be able to move between organs and within tissues. Recent advances in fluorescent reporter technology and light microscopy have provided better images than ever before of the location and behaviour of leukocytes in complex 3D environments and also within different tissues in intact living organisms [1]. However, a significant bottleneck in converting these images into interpretable data is their reliable analysis. Fluorescently labelled cells must be tracked through time-lapse image-series data of varying contrast [2]. Often the most effective way to do this is by labour intensive and manual annotation, however, this seriously limits the number of cells that can be analysed (and thus the conclusions that may be drawn).

In order to enable quantitative analysis of a large number of cells, we propose a semi-automated processing pipeline based on state-of-the-art methods for background motion compensation, cell detection, and tracking. Figure 1 illustrates the pipeline components and their order of execution. Before describing each component in detail in Section 2, we first discuss related work in the area of time-lapse microscopic image processing. In Section 3, we demonstrate the performance of our approach on a set of challenging microscopic image sequences. We conclude the paper in Section 4 with a discussion about limitations and future work.

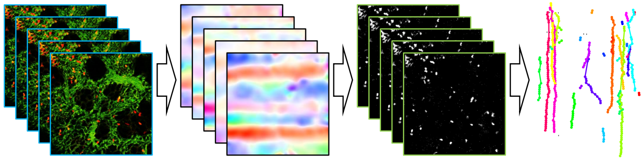


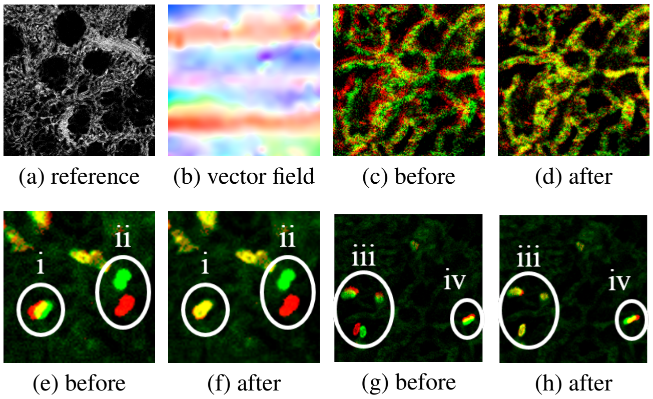
Fig. 1. Our processing pipeline for analysing time-lapse microscopic image sequences. First, we compensate for background motion induced by *in-vivo* laser scanning techniques and cardiac and respiratory activity. Then, cells are detected and tracked over time to obtain motion trajectories which enable the analysis of cell behaviour.

1.1. Related work

There is a substantial body of research into methods for cell detection and tracking, and a detailed survey is beyond the scope of this paper. An introduction to the topic and an overview of different methods and successes is provided by Meijering *et al.* [3]. Many approaches are tailored towards a particular imaging technique, and not directly applicable to novel sequences. To this end, we utilise two recently proposed approaches for detection [4] and tracking [5] which have been shown to generalise well and are applicable to data with varying characteristics. We modify and extend those methods and integrate them into our processing pipeline that has a motion compensation component as the first pre-processing step. A recently proposed motion compensation method [6] divides the image frames into segments and allows to compensate for translational motion between segments based on maximising correlation. While this method can effectively remove motion artifacts it necessitates the presence of a motion free period which might require modifications to the imaging protocol in terms of setting proper acquisition times [6]. In contrast, our method based on image registration can handle arbitrary, nonlinear background motion and does not impose particular constraints on the image acquisition. The details of the different processing components are described in the following.

2. METHOD

Our processing pipeline consists of three main components, motion compensation, cell detection, and tracking which are described the following subsections. The overall pipeline and how the components are connected is shown in Figure 1.



(a) reference (b) vector field (c) before (d) after

(e) before (f) after (g) before (h) after

Fig. 2. Illustration of the effect of background motion compensation. (a) Background channel used as reference image. (b) The (colour-coded) dense displacement field obtained when compensating the motion of the adjacent image frame. Motion artifacts caused by the laser scanning microscopy technique are clearly visible as streak-like patterns. In (c) and (d) close-ups of overlays of the reference frame and its adjacent frame before and after motion compensation. In (e)-(h) close-ups of overlays of the fluorescent image channels with leukocytes before and after motion compensation. Note, how the apparent motion of non-moving cells is compensated (i,iii), while actual cell motion is preserved (ii) and even corrected (iv). These observations are confirmed by visually inspecting subsequent frames and the corresponding locations of moving and non-moving cells.

2.1. Background motion compensation

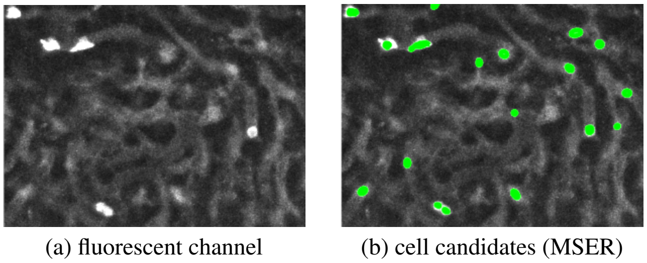
The first component of the processing pipeline is a fully-automatic background motion compensation which makes use of an efficient nonlinear registration method based on graph optimisation [7]. This component aims at removing non-cell related motion caused by cardiac and respiratory activity and the imaging procedure itself, e.g. from "line-by-line" techniques such as laser scanning microscopy. It is important to compensate for this "background motion", which might be otherwise mistakenly accounted for actual cell motion and could cause wrong conclusions about cell behaviour. An illustration of the effect of our motion compensation is shown in Figure 2.

We define a temporal sequence of microscopic images as $\mathcal{I} = \{I_t\}$ with I_t being the tth frame. Fluorescent imaging allows to simultaneously capture multi-channel images, where one of the image channels is explicitly used to capture background structure only (i.e, non-fluorescent tissue). We denote this background channel of frame t by I_t^b. For estimating the nonlinear motion between frames, we employ an intensity-based registration method using correlation coefficient (CC) as the similarity measure. The registration is performed on the background channels. Thus, we avoid the compensation of actual cell motion. We randomly select one frame of the sequence as the reference frame, denoted as I_r^b. For all other frames t, we then solve the optimisation problem

$$\hat{T} = \arg\min_{T} E(I_t^b \circ T, I_r^b) \quad , \tag{1}$$

where $T(\mathbf{x}) = \mathbf{x} + D(\mathbf{x})$ is a nonlinear spatial transformation of image coordinates $\mathbf{x}$ with a dense displacement field D. The energy function consists of the similarity measure and a regularisation term that favours smooth displacement fields:

$$E(I_t^b \circ T, I_r^b) = -S_{CC}(I_t^b \circ T, I_r^b) + \lambda\, R(T) \quad , \tag{2}$$



(a) fluorescent channel (b) cell candidates (MSER)

Fig. 3. First step of the cell detection component. In (a) is the input channel, and in (b) the MSER cell detections are overlaid in green.

where λ influences the amount of regularisation. The employed registration method makes use of free-form deformations as compact representation of the dense displacement field. The optimisation itself is performed using iterative multi-label graph-cuts, and for a pair of images of size 512×512 the registration takes only about 2 seconds. The estimated transformation is then used to warp the other (fluorescent) channels of the image frames. This results in a motion compensated multi-channel image sequence which is then passed to the next component of the processing pipeline.

2.2. Cell detection

Our cell detection component is a modified version of the recently proposed method by Arteta *et al.* [4]. The detector operates in three steps. First, a larger number of cell candidate regions is extracted from a fluorescent image using the Maximally Stable Extremal Region (MSER) detector [8]. This leads to a high recall, but relatively low precision. Figure 3 shows an example output of the MSER detector. In the second step, each of these regions R_i is assigned a probability $P(R_i)$ of being a cell obtained from a learned structured SVM classifier. The SVM can be trained on a relatively small number of training images (e.g., using the first few frames of a sequence) in which cells are annotated with single dots. In order to find the best set of features, we have evaluated the performance of the classifier on several combinations of features, namely the area of a region, intensity features, spatial locations, and shape. Overall best performance was obtained using the histogram of pixel intensities within the region, a shape descriptor of the boundary of the region, and a histogram of differences of intensities between the region border and a dilation of it. Based on the classifier output, the third step aims for selecting an optimal non-overlapping subset of candidates by maximising the sum of probabilities defined as

$$\hat{\mathbf{y}} = \arg\max_{\mathbf{y}\in\mathcal{Y}} \sum_{i=1}^{N} P(R_i)\, y_i \quad , \tag{3}$$

where y_i are indicator variables with $y_i = 1$ implying that region R_i is part of the subset, and thus, a detected cell. Here, $\mathcal{Y}$ is the space of non-overlapping subsets. In order to solve this problem efficiently via dynamic programming, the regions are organised in a tree structure according to a nestedness property. The exact details of this step can be found in [4].

2.3. Cell tracking

The last component of the pipeline takes the cell detections on every frame as input, and generates cell motion trajectories by linking de-

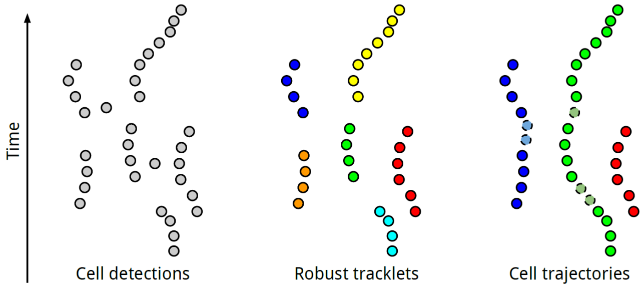


Fig. 4. The cell tracking works in two steps. First, detections that are very similar are linked into robust tracklets, which are then combined into longer trajectories. Gaps of several frames can be bridged.

tections between frames. It is important that the cell tracking component can deal with false positive and negative detections. Similar to the work by Bise *et al.* [5], our cell tracking method consists of two steps. First, cell detections are linked into *robust tracklets*, which are (shorter) sequences of cell detections between adjacent frames that can be linked with high confidence where no gaps are allowed. The second step then combines robust tracklets into (longer) sequences, or *trajectories*, using a linking model that allows to bridge between frames with missing detections. Figure 4 illustrates the two steps.

Linking detections into robust tracklets is based on evaluating the similarity between cell regions. To this end, we learn a naive Bayes classifier from example pairs annotated on the training frames. Negative examples are generated automatically by randomly sampling non-corresponding pairs. Only if the classifier predicts high similarity between two detections, a link is created. The similarity is based on the same set of features as used for the cell detector.

The process of linking robust tracklets into trajectories is then performed using a global data association approach which has been shown to perform favourably compared to alternatives [5]. The solution to the underlying maximum-a-posteriori (MAP) problem, which selects an optimal hypothesis of linking robust tracklets into trajectories, is computed using linear programming. Using Bayes' rule, the MAP problem can be rewritten as a product of probabilities

$$\begin{aligned}\mathbf{T}^* &= \arg\max_{\mathbf{T}} P(\mathbf{T}|\mathbf{X}) \\ &= \arg\max_{\mathbf{T}} \prod_{X_i \in \mathbf{X}} P(X_i|\mathbf{T}) \prod_{T_k \in \mathbf{T}} P_{traj}(T_k) \qquad (4)\end{aligned}$$

Here, $\mathbf{T}$ is a set of trajectories generated from the robust tracklets $\mathbf{X}$. Similar to [5], the likelihood of a robust tracklet X_i is defined as

$$P(X_i|\mathbf{T}) = \begin{cases} P_{TP}(X_i) & \text{if } \exists T_k \in \mathbf{T}, X_i \in T_k \\ P_{FP}(X_i) & \text{otherwise} \end{cases} , \qquad (5)$$

where $P_{TP}(X_i) = \alpha^{\frac{|X_i|}{\beta}}$ is the probability of being a true positive, $|X_i|$ is the number of detections, α corresponds to the miss detection rate, and β is a tuning parameter. The probability of false positives is then $P_{FP}(X_i) = 1 - P_{TP}(X_i)$. The prior probability of a trajectory T_k consisting of $\{X_j^k\}_{j=1}^n$ tracklets is defined as a Markov chain

$$P_{traj}(T_k) = P_{init}(X_1^k) \left[\prod_{j=1}^{n-1} P_{link}(X_{j+1}^k|X_j^k) \right] P_{term}(X_n^k) \ . \qquad (6)$$

Here, P_{init} and P_{term} are the probabilities of tracklets being the first and the last elements of a trajectory, and P_{link} is the probability of linking two tracklets having distance $\Delta(X_i, X_j)$, defined as

$$P_{link}(X_i|X_j) = \begin{cases} V(X_i, X_j) & \text{if } \Delta(X_i, X_j) \leq \delta_{max} \\ 0 & \text{otherwise} \end{cases} . \qquad (7)$$

The similarity $V(X_i, X_j)$ is learned using a binary neural net with Bayesian regularisation. To this end, a set of spatio-temporal and visual features are computed for pairs of tracklets. The features include the cell region descriptor used in the detection component, spatial information such as location and orientation of tracklets, and temporal distance (such as gap size and velocity) between start and end points of tracklets. The parameter δ_{max} limits the gap size that is considered for bridging over frames with missing detections. Based on P_{link}, we define P_{init} and P_{term} as

$$P_{init|term}(X_i) = \begin{cases} 1 - \max P_{link}(X_i|X_j) & \forall X_j \in \mathbf{X} \\ 0 & \text{otherwise} \end{cases} . \qquad (8)$$

which is equal to the probability of a tracklet X_i not being linked to its most similar tracklet. This allows trajectories to be initialised and terminate anywhere and at anytime in the sequence.

The advantage of the global data association approach is that all hypotheses over all frames of the sequence are considered simultaneously, rather than propagating the results from frame to frame. This makes the method robust to errors of the frame-by-frame cell detection component.

Datasets				Annotations				
Id	Tissue	Size	Frames	Frames	Cells	Traject.	Avg. C/F	Std. C/F
A	Liver	512 x 512	66	66	104	2	1.58	0.86
B	Liver	512 x 512	66	66	530	14	8.03	2.44
C	Lung	251 x 251	126	58	1128	7	19.45	4.19
D	Lung	199 x 199	377	53	547	5	10.32	2.54
E	Lung	277 x 277	194	67	470	7	7.01	2.25

Table 1. Details about the datasets with annotation statistics. The last two columns provide the ratios of cells per frame. Dataset A contains only a very few cells per frame. Our results indicate that this negatively affects the detection performance.

3. EXPERIMENTAL EVALUATION

We have tested our processing pipeline on five different microscopic image sequences with different imaging characteristics. Details about the number of frames, tissue type and annotation statistics per dataset are summarised in Table 1. In the following, we will present quantitative and qualitative results for different datasets.

3.1. Detection accuracy

In order to evaluate the accuracy of the cell detector, we used 70% of the annotations for training, and 30% for testing. Accuracy is quantified via precision, recall, and F1-score. Besides training and testing on frames from the same sequence, we were also interested in the generalisation performance when training and testing sequences do not overlap. This would enable a fully-automatic system where training is done only once. The quantitative results are summarised in Table 2. The F1-scores for training and testing on frames from the same sequence are 0.24, 0.90, 0.80, 0.86, 0.85, respectively for datasets A-E. Those increase for A to 0.54 and E to 0.88 when training is performed on the combined set ABCE. The F1-scores for the

Training Set	Precision					Recall				
	A	B	C	D	E	A	B	C	D	E
Individual	0.25	**0.90**	**0.77**	**0.86**	0.93	0.23	**0.89**	**0.84**	**0.85**	0.79
ABCDE	0.49	0.76	0.85	0.69	0.95	0.58	0.91	0.57	0.99	0.80
ABCD_	0.49	0.77	0.86	0.70	**0.93**	0.58	0.90	0.59	0.98	**0.81**
ABC_E	**0.46**	0.75	0.83	**0.41**	**0.87**	**0.65**	0.96	0.74	**0.99**	**0.90**
AB_DE	0.48	0.76	**0.86**	0.73	0.97	0.51	0.89	**0.46**	0.97	0.68
A_CDE	0.41	**0.63**	0.86	0.71	0.95	0.58	**0.89**	0.57	0.99	0.80
_BCDE	**0.49**	0.76	0.86	0.72	0.94	**0.58**	0.90	0.56	0.97	0.76

Table 2. Detection accuracy on five different datasets and combinations of training and testing. Numbers in red correspond to the highest F1-score. Numbers in blue correspond to the results when training and testing frames are taken from different image sequences.

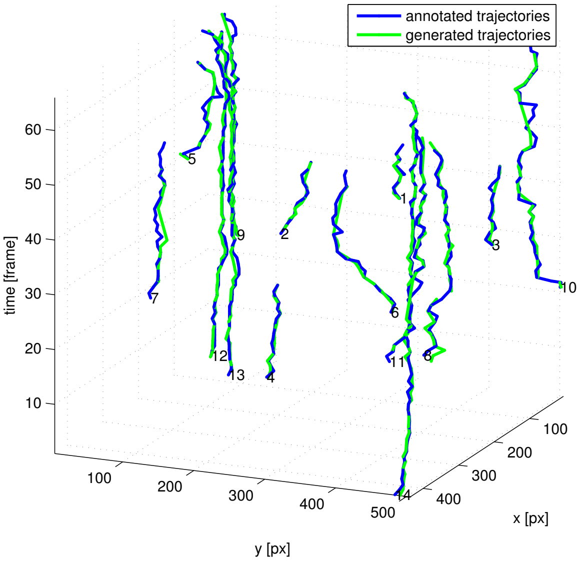


Fig. 5. Qualitative comparison of generated trajectories and their manual annotations for dataset B with 66 frames. Robust tracklets have been correctly linked into longer trajectories of cell motion.

case when training and testing sequences do not overlap are 0.53, 0.74, 0.60, 0.58, 0.87. The poor performance on dataset A is explained by a sudden contrast change within the sequence, and training on frames from other sequences improves the detection accuracy.

3.2. Tracking performance

To quantify the performance of the tracking component, we employ two metrics (also used in [5]), namely the *target effectiveness* (TE) and *track purity* (TP). TE corresponds to the percentage of frames in which a target (i.e., an annotation) is followed by a trajectory compared to the total number of target frames. Equivalently, TP corresponds to the percentage of frames in which a trajectory is followed by a target compared to the total number of trajectory frames. No tracking results were obtained for dataset A due to the poor detection performance. For datasets B-E the corresponding TE and TP values are 0.91/1.00, 0.93/0.88, 0.98/0.77, 0.94/0.93. A visual comparison between computed trajectories and manual annotations for dataset B is shown in Figure 5. The method can effectively bridge gaps between frames with false negative detection responses.

4. DISCUSSION

Our proposed processing pipeline combines state-of-the-art components for motion compensation, cell detection and tracking which achieves promising results on a set of different microscopic image sequences. An advantage of this modular approach is that individual components can be easily exchanged with either application-specific components (if needed) or with "updates" according to future advances in registration, detection and tracking methodology. Limitations such as the non-overlapping constraint, for example, could be addressed using a different detection procedure [9]. Next steps include an evaluation on longer sequences. We are currently acquiring *in-vivo* sequences over periods of up to 30 minutes with several thousand frames. This poses new challenges in terms of robustness and efficiency for the employed processing methods. In order to facilitate future research and comparison with other approaches, the source code of our methods will be made publicly available.

5. REFERENCES

[1] Ronald N Germain, Mark J Miller, Michael L Dustin, and Michel C Nussenzweig, "Dynamic imaging of the immune system: progress, pitfalls and promise," *Nature Reviews Immunology*, vol. 6, no. 7, pp. 497–507, 2006.

[2] Leo M Carlin, Efstathios G Stamatiades, Cedric Auffray, Richard N Hanna, Leanne Glover, Gema Vizcay-Barrena, Catherine C Hedrick, H Terence Cook, Sandra Diebold, and Frederic Geissmann, "Nr4a1-dependent Ly6C^{low} monocytes monitor endothelial cells and orchestrate their disposal," *Cell*, vol. 153, no. 2, pp. 362–375, 2013.

[3] Erik Meijering, Oleh Dzyubachyk, Ihor Smal, and Wiggert A van Cappellen, "Tracking in cell and developmental biology," *Seminars in cell & developmental biology*, vol. 20, no. 8, pp. 894–902, 2009.

[4] Carlos Arteta, Victor Lempitsky, J Alison Noble, and Andrew Zisserman, "Learning to detect cells using non-overlapping extremal regions," in *International Conference on Medical Image Computing and Computer Assisted Intervention*, N. Ayache, Ed. MICCAI, 2012, Lecture Notes in Computer Science, pp. 348–356, Springer.

[5] Ryoma Bise, Zhaozheng Yin, and Takeo Kanade, "Reliable cell tracking by global data association," in *Biomedical Imaging: From Nano to Macro, 2011 IEEE International Symposium on*. IEEE, 2011, pp. 1004–1010.

[6] Sungon Lee, Claudio Vinegoni, Matthew Sebas, and Ralph Weissleder, "Automated motion artifact removal for intravital microscopy, without a priori information," *Scientific Reports*, vol. 4, no. 4507, pp. 1–9, 2014.

[7] Ben Glocker, Nikos Komodakis, Georgios Tziritas, Nassir Navab, and Nikos Paragios, "Dense image registration through MRFs and efficient linear programming," *Medical image analysis*, vol. 12, no. 6, pp. 731–741, 2008.

[8] Jiri Matas, Ondrej Chum, Martin Urban, and Tomás Pajdla, "Robust wide-baseline stereo from maximally stable extremal regions," *Image and vision computing*, vol. 22, no. 10, pp. 761–767, 2004.

[9] Carlos Arteta, Victor Lempitsky, J Alison Noble, and Andrew Zisserman, "Learning to detect partially overlapping instances," in *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*. IEEE, 2013, pp. 3230–3237.