

Interactive Tracking of Cells in Microscopy Image Sequences

M. Gentil, M. Sameki, D. Gurari, E. Saraee, E. Hasenberg, J. Y. Wong, and M. Betke

Boston University, USA

Abstract. Analysis of the morphological changes that live cells undergo is an important task for cell biologists. Manual segmentation of images is extremely time consuming due to the large numbers of cells typically observed in an experiment. Automated methods are designed for specific datasets with suitable lighting and background conditions and make mistakes. We here propose a method to interactively track and segment living cells in time-lapse phase-contrast and fluorescence microscopy image sequences. Automated tools are used as much as possible to outline and track cells, and expert interaction is only needed when the algorithm performance is unsatisfactory. Our experiments show that the proposed system tracks 87 cells automatically, while recruiting expert involvement 66 times to re-initialize the algorithm when problems occur. It compares favorably with the baseline method that fails to track 35 cells for 1,423 frames of 2,824 frames in which they appear.

1 Introduction

The tasks of identifying cell boundaries and tracking them through time-lapse sequences are difficult when the cells divide, collide, or otherwise interact with neighboring cells [1]. It is also difficult to segment and track cells that undergo significant appearance variation in short periods of time or partially leave the field of view of the microscope camera (Figure 1). The research community has focused on developing automated methods for cell segmentation [2–5] and cell tracking [6–10], some of which attempt to maintain cell outlines when cells interact with each other [6, 10]. Mitosis, in particular, is one of the most challenging scenarios for cell tracking algorithms and large amount of research has been done to address the task of tracking new born cells [11–15].

One solution to overcome the difficulties of cell segmentation and tracking is to apply fully-automated approaches. Dzyubachyk et al. [7] presented an algorithm for multi-cell segmentation and tracking based on the coupled-active-surfaces algorithm. They proposed several modifications and extensions to improve robustness and applicability of this algorithm. Bise et al. [16] proposed a cell tracking method based on partial contour matching using dynamic programming. Taking the deformation of the cluster boundary contour as a cue, they matched optimal cell contours with cluster boundary contours. Although popular for reportedly having high accuracy and precision, these methods may not be generic since they have only been shown to work for specific datasets. It was therefore important that Chenouard et al. [17] performed a study that compares various cell tracking methods. They concluded that no universally best method exists for cell tracking.

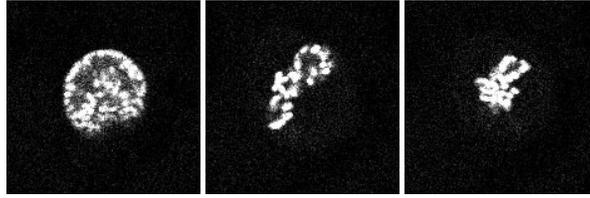


Fig. 1: One example a challenging case for cell segmentation where a significant variation of the cell shape is observed in consequent frames as a result of mitosis happening.

Our approach does not rely on automated segmentation and tracking algorithms alone but incorporates expert annotations in the process. The methodology of interaction has been embraced by others to address the difficulties of live cell tracking [18, 19]. Amat et al. [19], for example, proposed a method that computes a confidence score for the tracking result so that an expert can be involved in refining cell lineages when mistakes most probably occurred.

The research questions in regards to interactive cell tracking are to determine when automated algorithms fail and interaction is needed to ensure accurate tracking and how to limit the manual annotation efforts of experts. We employ automated approaches that are suitable for a variety of datasets and then compensate for errors by involving human interaction in the process.

We show that our method is general by testing it on six diverse datasets that exhibit non-optimal conditions for tracking algorithms such as variable lighting conditions, background noise and unsteady microscopes/cameras/substrates. We also demonstrate that the success of cooperation between machine and expert does not depend on the specific employed segmentation algorithm by using three different level set approaches. The concept of interactive cell tracking is indeed applicable to any state-of-the-art segmentation method.

Finally we show that a prediction model can be trained in order to detect when the algorithms fail and what kind of mistake was made during the automatic tracking. As a consequence, suitable help can be requested and experts can be involved in order to fix the incorrect or inaccurate boundaries drawn by the algorithms so that accurate tracking can continue.

2 Method

2.1 Automated Tracking

In order to track several cells during the duration of the image sequences, we use popular level set methods. We employ the Caselles Level Set algorithm [20] to segment cells that show a well-defined edge over a dark background and a Chan-Vese active contours approach [21] for sequences featuring less distinct cell boundaries. We implemented a hybrid tracking method that combines both of the above-mentioned algorithms depending on the average grayscale value of a cell. Specifically, we found that the edge-based method confuses cells with a gray-level contrast of 13 or lower (where 0 is black and

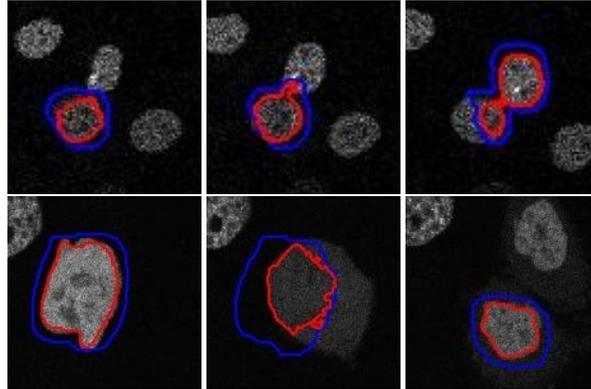


Fig. 2: Failures of the automated algorithm. Top: Two cells move close to each other and the algorithm merges their boundaries. Bottom: A cell undergoes mitosis, and the algorithm only tracks one of the daughter cells.

255 is white) to their background; as a result the tracking boundary is lost. Therefore in those cases we employ the Chan-Vese algorithm instead, otherwise we keep relying on the Caselles approach.

In order to initialize the level set methods, the expert manually annotates loose boundaries for cells in the first frame of each image sequence. This can be done with the LabelMe interface [22]. A tight boundary for each cell is then automatically produced by the level set segmentation algorithm in the first frame. In subsequent frames, the boundary of a cell in the previous frame is used as a mask to facilitate the detection of the boundary of the same cell in the current frame. Specifically, the previous boundary is first dilated by 20 pixels to draw the initial contour of a cell in the current frame, and then the level set method iteratively finds a tighter fitting contour around the cell.

While this method holds for tracking cells in well lighted and sparsely populated environments, it fails when dealing with challenging datasets such as the ones employed in our studies. In particular, we address 3 different situations that usually pose problems for accurate tracking:

- **Merging boundaries:** If two cells that are very close to each other collide, their boundaries are merged and the segmentation algorithm fails, as shown in Figure 2, top.
- **Mitosis:** Those cells which undergo mitosis need to be tracked carefully as they evolve their shape and luminosity very quickly and unexpectedly, as illustrated in Figure 2, bottom; therefore algorithms often fail to correctly keep track of the two newborn cells and re-initialization of boundaries for the two daughters is needed.
- **Expanding boundaries:** In some cases, a cell that is being tracked moves completely out of its initial boundary as a result of a sudden microscope, camera, or substrate movement, or because the cell exits the field of view of the microscope. Consequently, the segmentation method is not able to find any cell regions inside the mask and starts expanding the boundary indefinitely.

Next we run the tracking for the six selected datasets and an expert is asked to oversee the tracking output. This means the expert must watch the tracking of each cell boundary. When the expert observes a mistake during the process, the algorithm is stopped and the expert draws new boundaries for cells that require it by using the LabelMe annotating tool [22]. The involvement of an expert during the whole duration of the tracking is expensive, time-consuming, and tiring. It is preferable that the system automatically determines when a mistake is made so that the expert only needs to correct mistakes, and not watch the entire image sequence. We next describe how this can be accomplished with a prediction scheme.

2.2 Feature Extraction for Predicting When Algorithms Fail

We observe that the boundaries of cells that are not tracked correctly usually undergo significant changes like variations in pixel area, circularity and mean grayscale value. These properties can be used in order to automatically predict when the tracking algorithm is making a mistake and to only recruit expert intervention when a failure is detected.

The following features for image regions are meaningful to expose algorithm shortcomings:

- **Area ratio:** For every cell in the n -th frame of an image sequence, the area ratio

$$A_{r,n} = \frac{A_n}{\frac{1}{n+1} \sum_{k=0}^n A_k},$$

is computed, where A_n is the cell area in the n -th frame. This metric is very useful to detect sudden changes in the size of a cell which are often associated to an uncommon event, such as mitosis, which is observable in Figure 2, bottom.

- **Circularity:** Circularity

$$C_n = \frac{4\pi A_n}{P_n^2},$$

where P_n is the cell perimeter in the n -th frame, is useful to help the predictor distinguish between different type of cells as they may exhibit varied behavior for the same biological process, e.g. mitosis.

- **Circularity ratio:** The circularity ratio

$$C_{r,n} = \frac{C_n}{\frac{1}{n+1} \sum_{k=0}^n C_k},$$

circularity ratio is useful for detecting sudden fluctuations of the shape of a cell that can be caused by an incorrect cell boundary.

- **Grayscale Ratio:** The average grayscale ratio

$$G_{r,n} = \frac{G_n}{\frac{1}{n+1} \sum_{k=0}^n G_k},$$

where G_n is the cell mean grayscale value in the n -th image of the sequence, is useful to detect when a cell leaves its initial boundary. As mitosis usually implicates a lower cell luminosity, the grayscale ratio proves to be a relevant metric in order to identify this phenomenon.

2.3 Prediction Model

Our method classifies each cell in each frame of our image sequence as belonging to one of the following classes (see 2.1):

- Class 0: Correct tracking
- Class 1: Merging boundaries
- Class 2: Mitosis
- Class 3: Expanding boundary

As cell tracking is being performed, our predictor determines the class of every boundary and stops the tracking whenever one of the classes 1, 2 or 3 is obtained. Then an expert is asked to re-initialize the boundary for the cell that caused the automated method to fail according to the detected class: For class 1 two different boundaries for two standalone cells should be drawn. For class 2, the annotator needs to draw the loose boundaries of the two newborn cells. For class 3, a single annotation of the correct boundary is required.

In order to train a classification model to distinguish the 4 cell classes we used fully automated tracking on 11 fluorescence image sequences. Next we obtained ground truth class labels for each automatically tracked cell and extracted the 4 features mentioned above, thus building a training set of 5,159 samples.

Once we had features extracted for each of the four classes, we needed a simple classification method to label the unpredictable behavior of the cells in each frame based on these extracted features. We used a K-nearest neighbor method to label the cells based on the label of K nearest neighbors in the feature set [23].

We chose $K = 5$, thus the algorithm does not solely rely on the closest neighbors but it can make a more informed decision using a higher number of neighbors.

Majority Voting of Class Labels Due to natural correlation between features and their labels in the consequent frames, we applied majority voting to address this correlation and improve our results when the algorithm gets confused. The size of the window for our voting is three.

The most challenging case for our algorithm is to distinguish between the correct tracking of the cells and when they undergo mitosis, since it is not possible to exactly state at which frame mitosis is finished and cells are back to normal. This leads to confusion of the classification algorithm.

In order to overcome this issue, we increased the cost function of the misclassification of these two classes in K-nearest neighbors. This balances the misclassification among different classes that can be further improved by majority voting. It should be noted that if the number of misclassifications for each class is higher than a threshold, majority voting might accentuate the misclassification instead of improving the results. Changing the cost of misclassification based on the nature of the problem is therefore beneficial.

2.4 Evaluation Methodology

To collect expert-drawn segmentations, we used the freely-available online image annotation tool LabelMe [22]. We used the Jaccard index to measure the overlap of expert-

drawn and computer-drawn cell regions. Given a cell region A , drawn by an expert, and a cell region B , tracked by the computer, the Jaccard index computes the ratio of the number of pixels common to A and B over the number of pixels that are in at least one of the regions: $\frac{A \cap B}{A \cup B}$.

Given the large size of our dataset, we used a sampling scheme to select 36 images for which we acquired expert-drawn segmentations of all the cells present in the image. This yielded a dataset with 434 expert-annotated cell outlines. Note that the expert did not have the benefit of seeing subsequent frames, which would have helped to identify mitosis.

3 Experiment and Results

3.1 Data Sets

We used both phase contrast and fluorescence microscopy image sequences of live cells. The phase-contrast images are part of an unpublished dataset. They show fibroblast cells of a mouse strain recorded with a Zeiss Axiovert S100 microscope and Princeton Instruments 1300YHS camera every 30 seconds, resulting in 1,948 image sequences. The length of the sequence that we used ranges from frames 1,500 to 1,948. Figure 3(f) was taken from this dataset.

The fluorescence microscopy images were obtained from the 2013 Cell Tracking Challenge ¹ and they consist of four different kinds of cells. The two sequences illustrated in Figures 3(d) and 3(e) show rat mesenchymal stem cells on a flat polyacrylamide substrate observed with a PerkinElmer UltraVIEW ERS microscope every 20 minutes for a total of 48 frames each. Two datasets feature Chinese Hamster Ovarian nuclei overexpressing GFP-PCNA recorded with a Zeiss LSM 510 microscope with a time step of 9.5 minutes, resulting in 92 frames each (Figure 3(a)).

Two more arrays of images show GFP-GOWT1 mouse stem cells, monitored by a Leica TCS SP5 microscope every 5 minutes for an overall amount of 92 frames, see Figures 3(b) and 3(c). Lastly, five more datasets were involved only for training purposes; they display simulated nuclei moving on a flat surface.

3.2 Results

We showed that the tracking algorithms supervised by an expert were able to outline all cells in our dataset, even in challenging situations. Intervention from the expert was required in 66 frames out of a total of 872 images in order to track 87 cells.

To evaluate our tracking system, we selected six frames from each of six datasets and collected the average Jaccard score reported in Table 1. The datasets shown in Figures 3(a), 3(b), 3(c) and 3(f) yielded good results as their Jaccard scores are above 0.65 and the tracked outlines and the gold standard masks mostly overlapped.

¹ http://www.codesolorzano.com/celltrackingchallenge/Cell_Tracking_Challenge/Welcome.html

Table 1: Performance of the tracking algorithms supervised by the expert measured via the Jaccard metric. Datasets are labeled with letters according their images in Figure 3

	(a)	(b)	(c)	(d)	(e)	(f)
Jaccard	0.677	0.722	0.862	0.312	0.428	0.660

Interestingly, in several cases, the algorithm was able to keep track of cells undergoing mitosis while the expert, who only viewed single images and did not have access to preceding and subsequent images as the algorithm did, interpreted them as noise clusters. Examples are shown at the bottom of Figure 6(a), near the bottom right corner of Figure 6(b) and close to the top left corner of Figure 6(c).

For the datasets displaying rat stem cells, Figures 3(d) and 3(e), the algorithm produced less accurate segmentation results as their Jaccard scores are lower than 0.5. The inaccuracy of the cell boundaries is due to the noisiness of those image sequences and to the elusive nature of those cells as they do not show precise boundaries and even experts are unsure about their edges. Nonetheless, our method was able to keep track of the approximate cell position during the whole time-lapse microscopy video.

In order to expose the cases that cause automated algorithms to fail, we constructed a classifier based on K-nearest neighbors. To train our prediction model, we extracted a dataset of 5,159 cell regions for which we obtained the features described in 2.2 and ground truth labels for their class membership as defined in 2.3. Then we selected 684 windows of three subsequent frames for a random cell of our dataset and built a testing set with the features for it, resulting in 2,052 samples used for testing. The remaining 3,107 features were used to train the K-nearest neighbors with a parameter of $K = 5$ neighbors. Next, we applied the model on the testing dataset and exploited majority voting over the window of three consequent frames to get the final predicted class for a given cell. This approach resulted in the confusion matrix reported in Table 2. The overall accuracy of the prediction model is 96.2% which is an overall satisfactory performance even though our classifier encounters difficulties when recognizing mitosis as its recall metric for that particular class is only 62.2%. However our data is sequential so it is unlikely that our method would misclassify a cell undergoing mitosis for the whole duration of the process which usually lasts for several frames.

4 Discussion and Conclusions

Our results suggest that the concept of interaction between algorithms and expert annotation is valuable. It enables us to trade-off accuracy and cost, and achieve a correct tracking result with little effort from the expert. Our results also show that it is possible to automatically detect when tracking algorithms make mistakes by constructing and training a suitable classifier. Consequently, re-initialization of the algorithm by the expert is only required when the classifier indicates so. Our method is general in the sense that it was shown to work for different region-based tracking algorithms by involving

Table 2: Confusion matrix for the K -nearest neighbors prediction model with majority voting. Ground truth labels for the testing set (columns) and predicted values for each of the 4 classes (rows) are reported. The last row and column show the recall and precision scores for each class; the bottom right cell reports the overall accuracy of the model.

	GT 0	GT 1	GT 2	GT 3	Prec
Pred 0	586	2	10	1	97.8%
Pred 1	3	13	0	1	76.5%
Pred 2	2	0	23	0	92.0%
Pred 3	2	1	4	36	83.7%
Recall	98.8%	81.3%	62.2%	94.7%	96.2%

different active contours models in the tracking. We suggest that our method will work for many different types of data since we successfully tested it on very diverse image sequences and non-optimal conditions for algorithms.

For future work, we will improve our prediction model by providing more features and adding a new class for the detection of cells that enter the scene during the tracking. We will extend our classifier to more datasets in order to improve its performance and generalize it to handle even more complicated tracking situations.

Since the inclusion of experts is typically expensive, crowdsourcing has been considered for cell segmentation. Sameki et al. [24] showed how to accurately and efficiently collect cell outlines from crowd workers. In future work, we will consider involving the crowd in our interactive method, given that paid crowd-workers without domain-specific backgrounds have been shown to be reliable annotators of biomedical images [25].

Acknowledgments. The authors thank NSF (1421943) for partial support of this work.

References

1. J. Rittscher. Characterization of biological processes through automated image analysis. *Annual Review of Biomedical Engineering*, 12:315–344, 2010.
2. I. Ersoy, F. Bunyak, M. A. Mackey, and K. Palaniappan. Cell segmentation using hessian-based detection and contour evolution with directional derivatives. In *Proc. of the Intl. Conf. on Image Processing (ICIP)*, pages 1804–1807, San Diego, 2008.
3. F. Li, X. Zhou, H. Zhao, and S. T. C. Wong. Cell segmentation using front vector flow guided active contours. In *Proc. of the Intl. Conf. on Medical Image Computing and Computer Assisted Intervention (MICCAI)*, pages 609–616, 2009.
4. S. K. Nath, K. Palaniappan, and F. Bunyak. Cell segmentation using coupled level sets and graph-vertex coloring. In *MICCAI*, pages 101–108, 2006.
5. L. Liu and S. Sclaroff. Medical image segmentation and retrieval via deformable models. In *ICIP*, pages 1071–1074, 2001.
6. R. Bise, K. Li, S. Eom, and T. Kanade. Reliably tracking partially overlapping neural stem cells in dic microscopy image sequences. In *MICCAI Workshop on Optical Tissue Image Analysis in Microscopy, Histopathology and Endoscopy (OPTMHIsE)*, 2009.

7. O. Dzyubachyk, W. A. van Cappellen, J. Essers, W. J. Niessen, and E. Meijering. Advanced level-set-based cell tracking in time-lapse fluorescence microscopy. *IEEE Transactions on Medical Image Processing*, 29(3):852–867, 2010.
8. D. House, M. L. Walker, Z. Wu, J. Y. Wong, and M. Betke. Tracking of cell populations to understand their spatio-temporal behavior in response to physical stimuli. In *MMBIA 2009: IEEE Computer Society Workshop on Mathematical Methods in Biomedical Image Analysis*, Miami, FL, June 2009. 8 pp.
9. E. Meijering, O. Dzyubachyk, and I. Smal. Methods for cell and particle tracking. *Methods Enzymol*, 504(9):183–200, 2012.
10. Z. Wu, D. Gurari, J. Y. Wong, and M. Betke. Hierarchical partial matching and segmentation of interacting cells. In *Proceedings of the 15th International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI), Nice, France, Oct., 2012*. 8 pp.
11. N. Harder, F. Mora-Bermúdez, W. J. Godinez, A. Wünsche, R. Eils, J. Ellenberg, and K. Rohr. Automatic analysis of dividing cells in live cell movies to detect mitotic delays and correlate phenotypes in time. *Genome research*, 19(11):2113–2124, 2009.
12. D. Padfield, J. Rittscher, N. Thomas, and B. Roysam. Spatio-temporal cell cycle phase analysis using level sets and fast marching methods. *Medical image analysis*, 13(1):143–155, 2009.
13. M. Held, M. Schmitz, Fischer, et al. Cellcognition: time-resolved phenotype annotation in high-throughput live cell imaging. *Nature Methods*, 7(9):747–754, 2010.
14. A. El-Labban, A. Zisserman, Y. Toyoda, A. W. Bird, and A. Hyman. Discriminative semi-Markov models for automated mitotic phase labelling. In *2012 9th IEEE International Symposium on Biomedical Imaging (ISBI)*, pages 760–763. IEEE, 2012.
15. S. Huh, S. Eom, L. Weiss, T. Kanade, et al. Mitosis detection of hematopoietic stem cell populations in time-lapse phase-contrast microscopy images. In *9th IEEE International Symposium on Biomedical Imaging (ISBI)*, pages 390–393, 2012.
16. R. Bise, K. Li, S. Eom, and T. Kanade. Reliably tracking partially overlapping neural stem cells in dic microscopy image sequences. In *Proc. (MICCAI) 2009: Workshop on Optical Tissue Image Analysis in Microscopy, Histopathology and Endoscopy (OPTMHIS)*, 2009.
17. N. Chenouard, I. Smal, F. De Chaumont, et al. Objective comparison of particle tracking methods. *Nature Methods*, 11(3):281, 2014.
18. M. B. Smith, E. Karatekin, A. Gohlke, et al. Interactive, computer-assisted tracking of speckle trajectories in fluorescence microscopy: Application to actin polymerization and membrane fusion. *Biophysical Journal*, 101(7):1794–1804, October 2011.
19. F. Amat, W. Lemon, D. P. Mossing, et al. Fast, accurate reconstruction of cell lineages from large-scale fluorescence microscopy data. *Nature Methods*, 11(9):951–958, July 2014.
20. V. Caselles, R. Kimmel, and G. Sapiro. Geodesic active contours. *International Journal of Computer Vision*, 22:61–79, 1997.
21. T. F. Chan and L. A. Vese. Active contours without edges. *IEEE Trans Image Process*, 10(2):266–277, 2001.
22. B. C. Russell, A. Torralba, K. P. Murphy, and W. T. Freeman. LabelMe: A database and web-based tool for image annotation. *Int J Comp Vis*, 77(1–3):157–173, 2008.
23. Richard O Duda and Peter E Hart. Pattern classification and scene analysis. *J. Wiley and Sons*, 1973.
24. M. Sameki, D. Gurari, and M. Betke. ICORD: Intelligent collection of redundant data – a dynamic system for crowdsourcing cell segmentations accurately and efficiently. In *CVPR Wrkshp. on Comput. Vision for Microscopy Image Analysis (CVMI)*, 2016. 10 pp.
25. D. Gurari, D. Theriault, M. Sameki, B. Isenberg, T. A. Pham, A. Purwada, P. Solski, M. Walker, C. Zhang, J. Y. Wong, and M. Betke. How to collect segmentations for biomedical images? A benchmark evaluating the performance of experts, crowdsourced non-experts, and algorithms. In *IEEE Winter Conf. on Appl. of Computer Vision (WACV)*, 2015. 8 pages.

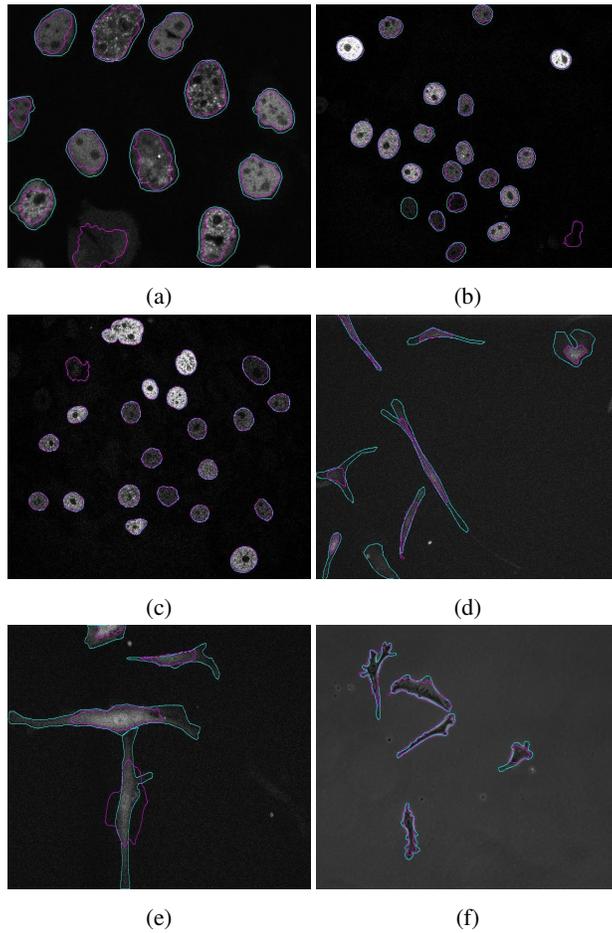


Fig. 3: Results of the tracking output on the 6 different datasets employed. The blue boundaries enclose the cell according to the ground truth labeling provided by the expert, while the pink outlines indicate the cell segmentations output by the algorithm. For datasets (a), (b), (c) and (f), the tracking is mostly correct. Datasets (d) and (e) were more challenging, as the cell edges are poorly defined and both the algorithm and the expert were unsure about the correct boundaries.