

# Automated and Semi-automated Cell Tracking: Addressing Portability Challenges (Supplementary Information)

*Andrey Kan, James Bailey, Christopher Leckie,  
John Markham, Mark R. Dowling, Rajib Chakravorty*

## A Assumption about the Cell Dynamics

We only consider the distributions of  $P(i \rightarrow j)$  with zero mean, because before doing any assignment we can subtract the superposition of offsets over all pairs of measurements, e.g., as proposed by Chowdhury et al. [2010].

Let  $P(i \rightarrow j)$  be the probability that a cell moves between the locations  $x_i$  and  $x_j$ ,  $\vec{x}_i, \vec{x}_j \in \mathbb{R}^n$ . This probability can be written as  $P(i \rightarrow j) = P(\vec{\theta})P(r^2|\vec{\theta})$ , where  $\vec{\theta}$  is a  $n$  dimensional vector of angles between the line  $\vec{x}_i\vec{x}_j$  and each of the coordinate axes, and  $r^2$  is the squared Euclidean distance  $r^2 = \|\vec{x}_i - \vec{x}_j\|^2$ .

We do not restrict our algorithm to any specific distribution of  $\vec{\theta}$ , and we estimate  $P(i \rightarrow j)$  only on the basis of the expected probability for the distance

$$P(i \rightarrow j) \propto \int_0^{2\pi} \dots \int_0^{2\pi} P(\vec{\theta})P(r^2|\vec{\theta}) d\theta_1 \dots d\theta_n. \quad (16)$$

As defined by Gomez et al. [1998], the  $n$  dimensional power exponential distribution with zero mean

$$P(\vec{x}) = k(n, \beta) |\Sigma|^{-\frac{1}{2}} \exp \left\{ -\frac{1}{2} [\vec{x}^T \Sigma^{-1} \vec{x}]^\beta \right\}, \quad (17)$$

where  $k(n, \beta)$  is a parameter that determines the shape of the distribution,  $\Sigma$  is a positive definite covariance matrix, and superscript  $T$  means the transpose of a matrix, e.g.,  $A^T$  is a transpose of matrix  $A$ .

Here  $\vec{x} \in \mathbb{R}^n$  can be written as  $\vec{x} = r \cdot \cos \vec{\theta}$ , and the expression  $\vec{x}^T \Sigma^{-1} \vec{x} = r^2 [(\cos \vec{\theta})^T \Sigma^{-1} (\cos \vec{\theta})]$ , where  $\cos \vec{\theta}$  denotes the  $n$  dimensional vector with elements  $\cos \theta_i$ .  $\Sigma$  is a positive definite matrix, hence  $\Sigma^{-1}$  is also a positive definite and thus  $[(\cos \vec{\theta})^T \Sigma^{-1} (\cos \vec{\theta})] > 0$  for all  $\vec{\theta}$ . We have that for all  $\vec{\theta}$ ,  $\vec{x}^T \Sigma^{-1} \vec{x}$  increases with  $r^2$  and thus expression 17 decreases with  $r^2$ .

## B Two Round Assignment Approach

In this section, we repeat some of the text (definitions and problem statement) from Section 2.2 for convenience of reading.

Suppose that at frame  $f$  we have a set of previously established tracks  $T_f = \{t_i\}$ , where a track  $t_i$  has the location  $m_{f,i}$ . Further, suppose that at frame  $f + 1$  we have a set of measurements  $M_{f+1} = \{m_{f+1,j}\}$ .

Consider assigning measurements in  $M_{f+1}$  to tracks in  $T_f$  such that a measurement  $m$  can be assigned to 0 or 1 tracks, and 0, 1, or 2 measurements can be assigned to a track  $t$ . Fig. 4 (in main text) explains the choice of this assignment setup with an example. We refer to an individual assignment of measurement  $j$  to track  $i$  as a *link*  $i - j$ . The link  $i - j$  represents a potential cell offset from the location  $m_{f,i}$  to the location  $m_{f+1,j}$ . The *length of a link*  $i - j$  is the Euclidean distance  $r_{ij} = \|\vec{x}_i - \vec{x}_j\|$  between the locations of  $m_i$  and  $m_j$ . Where unambiguous, we use term link to denote the length of a link.

We refer to a set of tracks, measurements, and established links between them as an *assignment*. Our task is to find the assignment that corresponds to the true set of cell events that occur between frames  $f$  and  $(f + 1)$ . A popular approach for solving this kind of task is based on Bayesian inference. Here the likelihood of an assignment is expressed using the probabilities of individual events implied by the assignment (Kirubarajan et al., 2001; Mori et al., 1992). The four possible events in the assignment are as follows.

1. A one-to-one assignment  $i - j$  (e.g., link 1 — 1 in Figure 4) implies a cell move between locations  $m_i$  and  $m_j$ , and we express the probability of such an event as  $P_m \cdot P(i \rightarrow j)$ .
2. A two-to-one assignment  $i - j$  and  $i - k$  (e.g., links 2 — 2 and 2 — 3 in Figure 4) implies the following set of events: {a cell division, a displacement between  $m_i$  and  $m_j$  of the first daughter cell, and a displacement between  $m_i$  and  $m_k$  of the second daughter cell}. We express the probability of this events as  $P_s \cdot P(i \rightarrow j) \cdot P(i \rightarrow k)$ .
3. A zero-to-one assignment (e.g., point 3 at frame  $f$  in Figure 4) implies a cell disappearance: death or a false negative error. The probability of this event is  $P_{dis} = P_d + P_{fn}$ .
4. A one-to-zero assignment (e.g., point 4 at frame  $f + 1$  in Figure 4) implies a false positive error with the probability  $P_{fp}$ .

Under the independence assumption 2, we can express the likelihood of assign-

ment  $A$  as

$$\begin{aligned}
L(A) &= \left[ \prod_{a \in M(A)} P_m P(i_a \rightarrow j_a) \right] \\
&\times \left[ \prod_{b \in S(A)} P_s P(i_b \rightarrow j_b) P(i_b \rightarrow k_b) \right] \\
&\times \left[ \prod_{c \in D(A)} P_{dis} \right] \times \left[ \prod_{d \in FP} P_{fp} \right] \tag{18}
\end{aligned}$$

Here,  $M(A)$ ,  $S(A)$ ,  $D(A)$  and  $FP(A)$  are the sets of links and measurements corresponding to a move, division (split), disappearance and false positive configurations respectively. At least one of the sets is non-empty. Now our problem is to find assignment  $A'$  that maximizes expression 18, that is, find any  $A' = \arg \max_A L(A)$ . In what follows, we shorten expressions like  $L(A)$  to  $L$ .

A straightforward solution of the problem (e.g., using integer programming) requires values for parameters  $P_s$ ,  $P_d$ ,  $P_{fp}$ , and  $P_{fn}$ . In what follows, we propose an approach that delivers an approximate solution, but does not require knowledge of the values for these parameters. The idea is to replace the four parameters by a single parameter, called the gating distance (introduced later), and then use our technique to estimate the value of this remaining parameter. Our approach is based on a simplification of the objective function (equation 18) in three steps.

### B.1 Simplification Step 1

The first simplification step is based on the following lemma.

**Lemma 3.** *Consider an arbitrary assignment, such that there exist links  $i - j$ ,  $i - l$  and unassigned track  $t_k$  (Fig. 12A). If  $P(i \rightarrow l) < 81 \cdot P(k \rightarrow l)$  then removing the link  $i - l$  and adding the link  $k - l$  (Fig. 12B) increases the likelihood of the assignment.<sup>1</sup>*

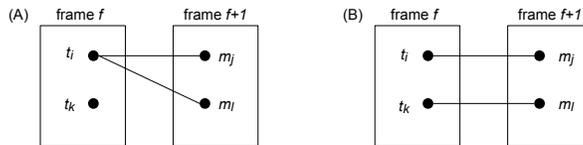


Fig. 12: (Fig. A1) Possible configurations of links between two measurements and two tracks. Lemma 3 states that configuration in the Figure B is more likely.

<sup>1</sup> Proofs of all lemmas are given in Appendix C.

For a random pair of move probabilities  $P(i \rightarrow l)$  and  $P(k \rightarrow l)$ , it is likely that  $P(i \rightarrow l) < 81 \cdot P(k \rightarrow l)$ , because the second term is multiplied by a large coefficient. The particular value here is not important (the number 81 arises from probabilities in Assumption 1, see the proof). Thus it is likely that the second configuration is favorable. As an approximation, we put that there cannot be divisions, unless all tracks have been assigned a measurement. Therefore, we propose to seek for an assignment that maximizes equation 18 using a two-round assignment approach, where, in each round, assigning two measurements to one track is not permitted. We then find optimal assignments in each round using modified objective functions, presented below, and then combine two found assignments into one (Fig. 13). The advantage of our two round approach is that the divisions are eliminated from a single assignment round. A division event occurs when there are measurements assigned to the same track in both rounds.

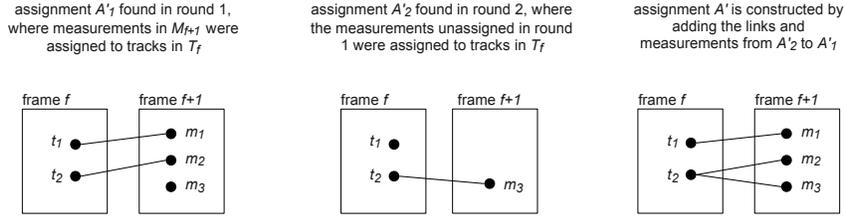


Fig. 13: (Fig. A2) Two-round assignment approach. In a single round two-to-one links are not permitted, hence division events are eliminated from a single round. However, a division event can occur when there are measurements assigned to the same track in both rounds.

Let  $A'_1$  be an optimal assignment in the first round, and  $A'_2$  be an optimal assignment in the second round. We approximate an optimal interframe assignment by the assignment  $A'$  that is derived by adding the links and measurements from  $A'_2$  to  $A'_1$  (Fig. 13).

## B.2 Simplification Step 2

We now present the objective functions for the two assignment rounds. Recall that the probability of a division event is  $P_s \cdot P(i \rightarrow j) \cdot P(i \rightarrow k)$ . This can be written as  $[P_m P_{fp} P(i \rightarrow j)] \cdot \left[ \frac{P_s P(i \rightarrow k)}{P_m P_{fp}} \right]$ . We use this decomposition to divide the likelihood in equation 18 into two parts  $L = L'_1 L'_2$ , where one part contains one of the terms with  $P(i \rightarrow j)$  or  $P(i \rightarrow k)$  for each division.

Using this decomposition, in the first round we assign measurements in  $M_{f+1}$

to tracks in  $T_f$ , and seek an assignment  $A'_1$  that maximizes

$$L'_1 = \left[ \prod_{a \in M'_1} P_m P(i_a \rightarrow j_a) \right] \times \left[ \prod_{b \in D'_1} P_{dis} \right] \left[ \prod_{c \in FP'_1} P_{fp} \right]. \quad (19)$$

Here  $M'_1$ ,  $D'_1$ , and  $FP'_1$  are the sets of links and measurements corresponding to the selection of moves, disappearances and false positives in the first round. Note that at least one of the sets is non-empty.

In the second round, we assign the measurements that are left unassigned after the first round (i.e., measurements in  $FP'_1$ ), to tracks in  $T_f$ . Here we seek an assignment  $A'_2$  that maximizes

$$L'_2 = \min \left[ 1, \prod_{a \in M'_2} \frac{P_s P(i_v \rightarrow k_v)}{P_m P_{fp}} \right]. \quad (20)$$

Here the measurements or tracks that are left unassigned imply disappearance or false positive events that have already been implied in the first round. Thus we omit the corresponding terms in equation 20.  $M'_2$  corresponds to the selection of links. Note that if  $M'_2$  is empty,  $L'_2 = 1$ .

**Lemma 4.** *Let  $A'_1$  and  $A'_2$  be optimal assignments in the first and the second rounds respectively. If there is a link  $i - k$  in  $A'_2$ , then there must be a link  $i - l$  in  $A'_1$ .*

This lemma shows that in the first assignment round we select the links that correspond to cell moves, and we also follow one of the daughters in divisions. In the second round, from this set of links we select those that correspond to cell divisions (we follow the second daughter).

### B.3 Simplification Step 3

Finally, we simplify both rounds further by introducing gating. Recall that each link  $i - j$  has a corresponding probability  $P(i \rightarrow j)$ . By *gating* we mean that we prohibit making the links that have  $P(i \rightarrow j) \leq P_g$ , where  $P_g$  is some probability gating threshold. We now want to express the gating threshold in terms of other parameters. If we set  $P_g = \sqrt{\frac{P_m P_{fp}}{P_s}}$  then the following property holds.

**Lemma 5.** *If the links that have  $P(i \rightarrow j) \leq \sqrt{\frac{P_m P_{fp}}{P_s}}$  are prohibited, then the optimal assignment in the first or second round has the maximum possible number of allowed links.*

Each link  $i - j$  has a length denoted  $r_{ij}$ , and under Assumption 3 about cell dynamics,  $P(i \rightarrow j)$  is decreasing with  $r_{ij}^2$ . Hence there exists some distance threshold  $R_g$  such that the links that have  $P(i \rightarrow j) \leq P_g$  also have  $r_{ij}^2 \leq R_g$  and vice versa. Threshold  $R_g$  is called *the gating distance*. Let  $R_g$  be a parameter of our tracking problem. Then in both assignment rounds, we can follow the same procedure: first, we prohibit the links where  $r_{ij} \geq R_g$ ; next, we only consider assignments with the maximum number of possible links; finally, we select an assignment  $A''$  that maximizes

$$L'' = \prod_{a \in \text{links}} P(i_a \rightarrow j_a), \quad (21)$$

or, taking the log likelihood, minimizes assignment cost

$$C = \sum_{a \in \text{links}} -\ln P(i_a \rightarrow j_a) = \sum_{a \in \text{links}} c(i_a \rightarrow j_a). \quad (22)$$

Here,  $c(i \rightarrow j)$  is the cost of link  $i - j$ . Under Assumption 3 about the cell dynamics, we propose to set  $c(i \rightarrow j) = r_{ij}^2$ , where  $r_{ij}^2$  is the squared Euclidean distance between locations  $m_i$  and  $m_j$ .

We have reduced our interframe association problem to the global nearest neighbor assignment task. This problem can be solved using the polynomial time algorithm of Munkres [1957]. After the simplification the solution does not require values of  $P_s$ ,  $P_d$ ,  $P_{fp}$ , and  $P_{fn}$ , that is we have replaced all the parameters from the assignment task by a single parameter  $R_g$ .

## C Proofs of Lemmas

*Proof of Lemma 1.* Equation 11 can be written as

$$P_f(r \leq R) + \frac{N_{all}}{\hat{N}_t} (P_{all}(r \leq R) - P_f(r \leq R)) = 1. \quad (23)$$

Let  $\hat{R}_{tmax}$  be the first point such that equation 23 holds, and let  $R_{fmax}$  be the first point such that  $P_f(r \leq R_{fmax}) = 1$ .

If  $\hat{R}_{tmax} \geq R_{fmax}$ , then  $P_f(r \leq \hat{R}_{tmax}) = 1$ . This implies  $P_{all}(r \leq \hat{R}_{tmax}) = P_f(r \leq \hat{R}_{tmax}) = 1$ , and thus  $\hat{R}_{tmax} \geq R_{tmax}$ .

If  $\hat{R}_{tmax} < R_{fmax}$  and  $\hat{R}_{tmax} < R_{tmax}$  then we have that  $P_f(r \leq \hat{R}_{tmax}) < 1$ , hence  $P_{all}(r \leq \hat{R}_{tmax}) > P_f(r \leq \hat{R}_{tmax})$ . We also have that

$$\begin{aligned} \frac{N_{all}}{\hat{N}_t} (P_{all}(r \leq \hat{R}_{tmax}) - P_f(r \leq \hat{R}_{tmax})) &> \\ \frac{N_{all}}{N_t} (P_{all}(r \leq \hat{R}_{tmax}) - P_f(r \leq \hat{R}_{tmax})). & \end{aligned} \quad (24)$$

This implies  $\hat{N}_t < N_t$  which contradicts the initial statement.

We have that in all possible cases, if  $\hat{N}_t \geq N_t$  then  $\hat{R}_{tmax} \geq R_{tmax}$ . □

*Proof of Lemma 2.* Consider a false negative error. An *error chain* is a set of false negative errors from the same track and from consecutive frames. The chain must contain at least 2 errors, and two consecutive chains from one track are considered to be one chain.

Let  $n$  be the total number of occurrences of all cells in all frames in a video. Then the number of false negative errors in the video is  $n_{fn} = P_{fn} \cdot n$  (defined by equation 5). It follows that the maximum number of chains for the video is  $\lfloor 0.5 \cdot n_{fn} \rfloor$ , and hence the probability of a false negative that starts a chain is less than  $\lfloor 0.5 \cdot P_{fn} \rfloor$ . Using assumption 1 about the detection quality, we have that the probability of a false negative starting a chain is less than 0.05.

Similar reasoning applies to false positive error. Here a chain is a set of false positive errors that are linked by NENIA. □

*Proof of Lemma 3.* Let  $A$  be an arbitrary assignment  $A$ , such that there exist links  $i - j$ ,  $i - l$  and unassigned track  $t_k$  (Fig. 12A). Here the two links and the track implies two events: a split of track  $t_i$ , and a disappearance of track  $t_k$ , with the total probability of the two events  $P_{dis} \cdot P_s \cdot P(i \rightarrow j) \cdot P(i \rightarrow l)$ .

Let  $B$  be the assignment derived from  $A$  by removing the link  $i - l$  and adding the link  $k - l$  (Fig. 12B). Here the division and disappearance events are replaced by two move events with the total probability  $P_m \cdot P_m \cdot P(i \rightarrow j) \cdot P(k \rightarrow l)$ .

We now compare  $P_{dis} \cdot P_s \cdot P(i \rightarrow j) \cdot P(i \rightarrow l)$  and  $P_m \cdot P_m \cdot P(i \rightarrow j) \cdot P(k \rightarrow l)$ , or equivalently, we compare  $P(i \rightarrow l)$  and  $\frac{P_m P_m}{P_s P_{dis}} P(k \rightarrow l)$ . Here  $P_{dis} = P_d + P_{fn}$ . Under assumption 1 about cell event probabilities we have  $P_{fn} \leq 0.1$  and  $P_s + P_d \leq 0.1$ , hence  $P_s \cdot P_{dis} \leq (0.1 - P_d)(0.1 + P_d) \leq 0.01$ . Under the same assumption we also have  $P_m \cdot P_m \geq 0.81$ , thus  $\frac{P_m P_m}{P_s P_{dis}} \geq 81$ .

We have that if  $P(i \rightarrow l) < 81 \cdot P(k \rightarrow l)$  then the likelihood of assignment  $B$  is greater than the likelihood of assignment  $A$ . □

*Proof of Lemma 4.* Let  $A'_1$  be an assignment that maximizes 19, and  $A'_2$  be an assignment that maximizes 20, and there is a link  $i - k$  in  $A'_2$ . Suppose, that there is no measurement assigned to track  $t_i$  in  $A'$ .

Note that  $\frac{P_s P(i \rightarrow k)}{P_m P_{fp}} \geq 1$ , otherwise link  $i - k$  would not be included in  $A'_2$ . This can be rewritten as  $P(i \rightarrow k) > \frac{P_m P_{fp}}{P_s}$ . Under assumption 1 about cell and detection events  $\frac{P_m P_{fp}}{P_s} > \frac{P_{dis} P_{fp}}{P_m}$ , hence  $P(i \rightarrow k) > \frac{P_{dis} P_{fp}}{P_m}$ .

We have that, in  $A'_1$ ,  $t_i$  and  $m_k$  were unassigned, but  $\frac{P_m}{P_m} \cdot P(i \rightarrow k) > P_{dis} \cdot P_{fp}$ . Hence adding a link  $i - k$  increases the likelihood of  $A'_1$ . But  $A'_1$  is an optimal assignment. We have a contradiction. □

*Proof of Lemma 5.* In this proof we denote any probability  $P(i \rightarrow j)$  as  $P(link)$ .

Consider an arbitrary assignment  $A_1$  in the first round after the gating. Suppose there is a potential link that is allowed, but not included in  $A_1$ . Including the link to  $A_1$  implies adding a move event and removing disappearance and

false positive events. Thus after the link inclusion the likelihood of  $A_1$  (equation 19) is multiplied by  $T_{11} = \frac{P_m P(link)}{P_{dis} P_{fp}}$ . Because the link is allowed, we have that  $P(link) > \sqrt{\frac{P_m P_{fp}}{P_s}}$ , hence  $T_{11} > \frac{P_m}{P_{dis}} \sqrt{\frac{P_m}{P_s P_{fp}}}$ , and, under assumption 1 about cell event probabilities,  $T_{11} > 1$ . Hence adding the link increases the likelihood of  $A_1$ .

Now suppose that there are no allowed links to add to  $A_1$ . Notice that removing any of the links already in  $A_1$  can introduce two other allowed links that can be added to  $A_1$ . Removing one link and adding two other links implies that the likelihood of  $A_1$  is multiplied by  $T_{12} = \frac{P_m P(link1) P(link2)}{P_{dis} P_{fp} P(link3)}$ . Note that  $T_{12} \geq \frac{P_m P(link1) P(link2)}{P_{dis} P_{fp}}$ . For any allowed link we have that  $P(link) > \sqrt{\frac{P_m P_{fp}}{P_s}}$ , and thus  $T_{12} > \frac{P_m P_m P_{fp}}{P_{dis} P_{fp} P_s}$ , and under assumption 1 about cell event probabilities,  $T_{12} > 1$ . Hence increasing the number of links increases the likelihood of  $A_1$ .

Consider an arbitrary assignment  $A_2$  in the second assignment round after the gating. Suppose there is a potential link that is allowed, but not included in  $A_2$ . Including the link to  $A_2$  implies that the likelihood of  $A_2$  (equation 20) is multiplied by  $T_{21} = \frac{P_s P(link)}{P_m P_{fp}}$ . Because the link is allowed, we have that  $P(link) > \sqrt{\frac{P_m P_{fp}}{P_s}}$ . This implies  $T_{21} > \sqrt{\frac{P_s}{P_m P_{fp}}}$ . If  $\frac{P_m P_{fp}}{P_s} > 1$  then  $P(link) > 1$  and there must be no allowed links, which contradicts the first statement in this paragraph. If  $\frac{P_m P_{fp}}{P_s} \leq 1$  then  $\sqrt{\frac{P_s}{P_m P_{fp}}} \geq 1$ , hence  $T_{21} > 1$ , and adding the link increases the likelihood of  $A_2$ .

Now suppose that there are no allowed links to add to  $A_2$ . Notice that removing any of the links already in  $A_2$  can introduce two other allowed links that can be added to  $A_2$ . Removing one link and adding two other links implies that the likelihood of  $A_2$  is multiplied by  $T_{22} = \frac{P_s P(link1) P(link2)}{P_m P_{fp} P(link1)}$ . Note that  $T_{22} \geq \frac{P_s P(link1) P(link2)}{P_m P_{fp}}$ . For any allowed link we have that  $P(link) > \sqrt{\frac{P_m P_{fp}}{P_s}}$ , and thus  $T_{22} > \frac{P_s}{P_m P_{fp}} \frac{P_m P_{fp}}{P_s} = 1$ . Hence increasing the number of link increases the likelihood of  $A_2$ . □

## D Median Filtering

In Section 2.3, we propose a method for reconstruction of the CDF of cell displacements. When using this method, the reconstructed CDF in general follows the true CDF curve, however, it is subject to noise (Fig. 14, left). Low pass filtering reduces the noise but introduces distortions of sharp edges in true CDF curves. Therefore, we use median filtering (Fig. 14, right).

Specifically, we use the filter implemented in MATLAB function *medfilt1*(.). This filtering algorithm has only one parameter (the window length). We make this algorithm parameter-free as follows. We set the parameter of the filtering to  $2W$ , where  $W$  is the maximum distance between two consecutive peaks (minimum or maximum) in the reconstructed CDF.

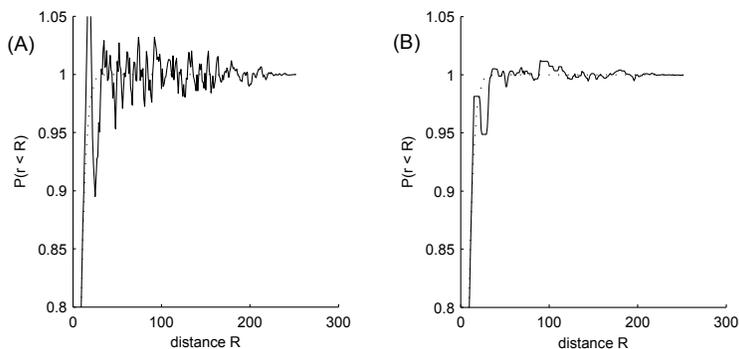


Fig. 14: (Fig. A3) Reconstruction of CDF of cell displacements results in a noisy curve (left). Median filtering reduces the noise (right). The dotted line represents the true CDF curve, the solid line is the reconstructed curve.

## E Details for Real Cell Videos

The differences in cell dynamics in our real videos were found as follows. We compared distributions of cell displacements between consecutive frames using a Kolmogorov-Smirnov (KS) test with the null hypothesis that the distributions are the same. Between each pair of videos, the null hypothesis was rejected at 5% significance level.

The normalized distributions of cell displacements (with zero mean and unit variance) also differed for all pairs of videos, except for pair *hex.6* and *square*, where the hypothesis was rejected at the 11% significance level. This comparison shows that the distributions of cell displacements vary in their shape.

Further, in our videos, we found that at least in four of them, cells are unlikely to follow Brownian motion. Brownian motion in 2 dimensions (i.e., a bivariate normal distribution of cell displacements) results in a Rayleigh distribution of the cell displacements [Papoulis, 1991]. Therefore, using the maximum likelihood method, for each video, we estimated the parameters of the Rayleigh distribution. For all videos, except *hex.6*, the distribution of cell displacements differed from the fitted Rayleigh distribution.

For each video, we generated detections with different qualities as follows. First, we manually produced the ground truth. Next, for a given detection quality as specified by  $\{P_{fp}, P_{fn}\}$ , we calculated the corresponding number  $n_{fp}$  of spurious measurements and the number  $n_{fn}$  of missed cell occurrences. We then randomly added  $n_{fp}$  spurious measurements and removed  $n_{fn}$  cell occurrences. For each video, and each pair  $\{P_{fp}, P_{fn}\}$  we generated 5 random versions of detections. For the level  $\{0, 0\}$  we generated 5 random variations of the ground truth, by adding normally distributed offsets to each cell occurrence in the ground truth.

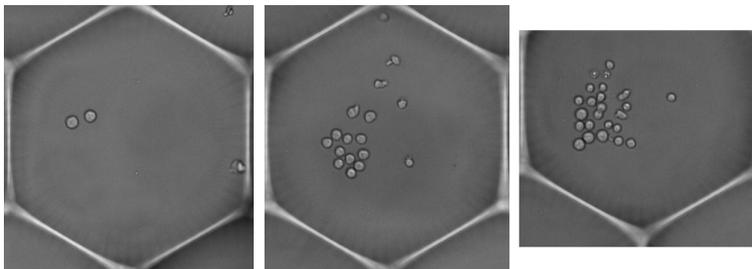


Fig. 15: (Fig. S1) Frames from videos (left to right) hex.6, hex.16, and hex.22. These videos have similar appearance, however we found differences in cell dynamics across the videos. Hence the same segmentation algorithm can be used, but the variability challenge still needs to be addressed by the tracker.

## F Evaluation on Synthetic Videos

In this section, we provide details for our synthetic evaluations presented in Section 3.3.

The synthetic videos of the first type consist of pairs of frames, where each pair is generated as follows. In the first frame, we placed  $n$  cells randomly with a uniform distribution in a square with side  $L$ . We then produced random offsets of these cells using the bivariate normal distribution with a diagonal covariance matrix having the same variance  $\sigma^2$  in each dimension. In the second frame, we placed cells at the offset positions of the cells in the first frame. A synthetic video is 100 random pairs with the same set of parameters  $\{n, \sigma, L\}$ .

For the evaluation of tracking performance scalability, we used fixed values  $L = 100$ ,  $P_{fp} = P_{fn} = 3\%$ , and took values of  $N$  (number of cells in each frame) in the range 30 – 200. We chose  $\sigma$  accordingly such that  $\beta_n = 3$ . We found that the tracking performance is similar for all  $N$  ( $P_{links} \approx 96\%$ ). We repeated the experiment for  $\beta_n = 10$  and varied  $N$  again. Again the observed performance was similar for all  $N$  ( $P_{links} \approx 86\%$ ). We found that, once the normalized density  $\beta_n$  is fixed, the tracking performance does not depend on the number of cells.

For the evaluation of the running time, we used fixed values  $L = 100$ ,  $P_{fp} = P_{fn} = 3\%$  and took values of  $N$  (number of cells in each frame) in the range 30 – 200. We chose  $\sigma$  accordingly such that  $\beta_n = 10$ . We run the algorithm on a desktop PC, with the gating distance set to infinity, that is we evaluated the worst case when all possible links needed to be considered. The running time was about 1.17 second per frame for  $N = 100$  cells, and 13 seconds per frame for  $N = 200$  cells. We found that the dependency of the running time on the number of cells can be fit with a degree 3 polynomial (mean root square error is about 0.02 seconds per frame, the data is not shown).

For testing the hypothesis about the tracking performance as a function of gating distance, we used fixed values  $N = 30$  and  $L = 100$ , and chose  $\sigma$  such

that  $\beta_n$  takes values in the range  $(0.01 \sim 100\,000) \times 10^{-2}$ , i.e., covers a much wider range of values than real videos. Recall that our real videos have  $\beta_n$  in the range  $(0.14 \sim 5.94) \times 10^{-2}$ .

For each video, we produced detections with quality levels  $P_{fp} = P_{fn}$  in  $\{0\%, 3\%, 5\%\}$ . For each detection quality, we calculated the required number of spurious observations and missed cell locations and distributed these errors uniformly across all frame pairs in the video. We then used NENIA to perform interframe assignments on each frame pair, and report an average performance  $P_{link}$  across all pairs for a given video with a given detection quality. We used NENIA with different gating distances  $R_g$ , to obtain the empirical performance curve.

All obtained empirical curves (see examples in Fig. 11 in main text, not all curves shown) preserve the prominent features of our hypothesized curve (Fig. 6 in main text): sharp rise from zero to maximum, maximum around the point  $R_g = R_{tmax}$ , and gradual decrease, after this point. We found that, as the detection quality decreases, the distance  $R_g$  that maximizes the performance shifts further away from  $R_{tmax}$ . This can be attributed to a higher proportion of false links compared to true links on a poorer detection quality. However, in the part of the curve around  $R_g = R_{tmax}$ , the exact value of  $R_g$  makes little difference. As a result, in all observed curves the difference  $P_{max} - P_{link}(R_{tmax}) < 2\%$ .

We evaluated the ability of NENIA to resolve cell divisions using synthetic videos of the second type generated as follows. Each synthetic video consists of 3 frames, where the first frame contains  $N_{cells} = 100$  cells randomly placed with a uniform distribution in the square with side  $L = 10,000$  units.  $N_{div} = 30$  random cells out of 100 divide and there are 130 cells in the second and third frames. Between each two frames a cell that does not divide moves with a random  $2D$  offset drawn from bivariate normal distribution with the same variance  $\sigma^2$  in each dimension. Cells that divide do not appear in the second frame. Instead, the two daughters of the divided cell appear. Each daughter is independently shifted from the mother cell’s position by a random  $2D$  offset drawn from bivariate normal distribution with the same variance  $\sigma^2$  in each dimension. Finally, a certain number of false positive measurements are added to the synthetic video. The number of false positives depends on the false positive rate parameter  $P_{fp}$ . We do not add false negative errors, because in this case some divisions can be missed due to missing cell locations rather than due to the abilities of NENIA.

In our evaluation, we fixed all parameters, except  $\beta_n$  and  $P_{fp}$ . Recall that  $\beta_n$  is the normalized object density that characterizes the “difficulty” of a video (Section 3.3). Using the definition of  $\beta_n$ , we put  $\sigma^2 = \frac{\beta_n \cdot L^2}{(N_{cells} + N_{div}) \cdot \pi}$ .

For each  $(\beta_n, P_{fp})$  pair, we generated 10 random variations of a synthetic video and ran NENIA with method B to resolve divisions. In Table 5 we report the mean and the standard deviation for ratios of correctly resolved divisions observed on 10 videos. We found that the ratio of correctly resolved divisions depends on both parameters  $\beta_n$  and  $P_{fp}$ . Further, we found that for every parameter combination, the majority of cell divisions are resolved correctly.

Tab. 5: Ratio of correctly resolved cell divisions for synthetic videos with different parameters  $\beta_n$  and  $P_{fp}$ . For every parameter combination, we tested 10 different synthetic videos, and report the mean and standard variation across 10 runs. For every parameter combination, the majority of cell divisions are resolved correctly.

	$P_{fp} = 0\%$	3%	5%
$\beta_n \times 10^2 = 0.1$	99.67; 1.05	88.67; 4.22	80.00; 6.48
1	95.00; 5.27	83.33; 4.16	80.67; 8.28
5	80.00; 6.08	71.00; 8.76	71.67; 6.33

Tab. 6: Number of frames used for SAMTRA training and the corresponding tracking performance gain

Number of Frames	20	40	60	80	100
Performance gain (%)	2	1	13	13	14

## G Evaluation of SAMTRA

In this section we present experiments that we conducted to find out how much training data is needed for SAMTRA to achieve a certain gain in tracking performance. The experimental setup was similar to that in Section 3.4 of the main text. The difference was that we varied the number of training frames from the *hex.22* video. For convenience, we first repeat the description of the experiment from the main text here.

In the evaluation, we used NENIA with method B, and we used  $P_{track}$  as the tracking performance measure. We trained SAMTRA using only one of the videos (*hex.22*) with one detection quality level ( $P_{fp} = P_{fn} \approx 2.5\%$ ). We then used all our videos at other detection levels (other values for  $P_{fp}$  and  $P_{fn}$ ) to evaluate SAMTRA. For each video, we randomly chose a detection such that the tracking quality on that detection is around 75%. We then randomly selected 10% of the frames and corrected the cell detection, where necessary, in the selected frames. We then ran the tracking again and recorded the tracking performance that we denote as  $P_{random}$ .

After that, we took the original detection and again selected 10% of frames. This time we selected frames with the highest probabilities of errors as predicted by SAMTRA. Where necessary, we corrected cell detection in these frames, ran tracking again, and recorded the tracking performance that we denote as  $P_{samtra}$ . The performance gain is defined as  $G = P_{samtra} - P_{random}$ . For different numbers of training frames and different testing videos we observed different gains.

In Table 6, we report a weighted average of the gain for a fixed number of training frames. A weight for a gain for a testing video is the number of correct tracks in that video divided by the number of correct tracks in all testing videos.

We conclude that a larger number of training frames tends to lead to a better

performance. At the same time, in our experiments, we observed a positive gain using as few as 20 frames. Note that correct tracks in our videos almost always span longer than 20 frames, and therefore in order to validate a single track manually one normally needs to review more than 20 frames. Therefore SAMTRA can be useful here.

We did not conduct experiments with more than 100 frames, because the *hex.22* video has only 100 frames (and we report the details of the evaluation on this level in Section 3.4 of the main text).

Finally, we note that as a rule of thumb the training can be considered to be satisfactory if the performance of the binary classifier (which is at the core of SAMTRA) is good. The performance of the classifier can be measured using standard techniques, such as cross-validation and calculating the area under receiver operating characteristic curve (ROC AUC). For example, a performance can be considered to be good when  $\text{ROC AUC} > 0.85$ .

## References

- A.S. Chowdhury, R. Chatterjee, M. Ghosh, and N. Ray. Cell Tracking in Video Microscopy using Bipartite Graph Matching. In *Proceedings of the 20th International Conference on Pattern Recognition*, pages 2456–2459, 2010.
- E. Gomez, M.A. Gomez-Vilegas, and J.M. Marin. A multivariate generalization of the power exponential family of distributions. *Commun. Stat. Theory Methods*, 27(3):589–600, 1998.
- T. Kirubarajan, Y. Bar-Shalom, and K.R. Pattipati. Multiassignment for tracking a large number of overlapping objects. *IEEE Trans. Aerosp. Electron. Syst.*, 37(1):2–21, 2001.
- S. Mori, K.C. Chang, and C.Y. Chong. Performance analysis of optimal data association with applications to multiple target tracking. In Y. Bar-Shalom, editor, *Multitarget-Multisensor Tracking: Applications and Advances*, volume 2, pages 183–235. Artech House, Boston, 1992.
- J. Munkres. Algorithms for the assignment and transportation problems. *J. of the Soc. for Ind. and Appl. Math.*, 5(1):32–38, 1957.
- A. Papoulis. *Probability, Random Variables and Stochastic Processes*. McGraw-Hill, New York, 3rd edition, 1991.