nature | **methods**

Robust single-particle tracking in live-cell time-lapse sequences

Khuloud Jaqaman, Dinah Loerke, Marcel Mettlen, Hirotaka Kuwata, Sergio Grinstein, Sandra L Schmid & Gaudenz Danuser

Supplementary figures and text:

Note: Supplementary Videos 1–3 and Supplementary Software are available on the Nature Methods website.

Gap closing, merging and splitting statistics for experimental data
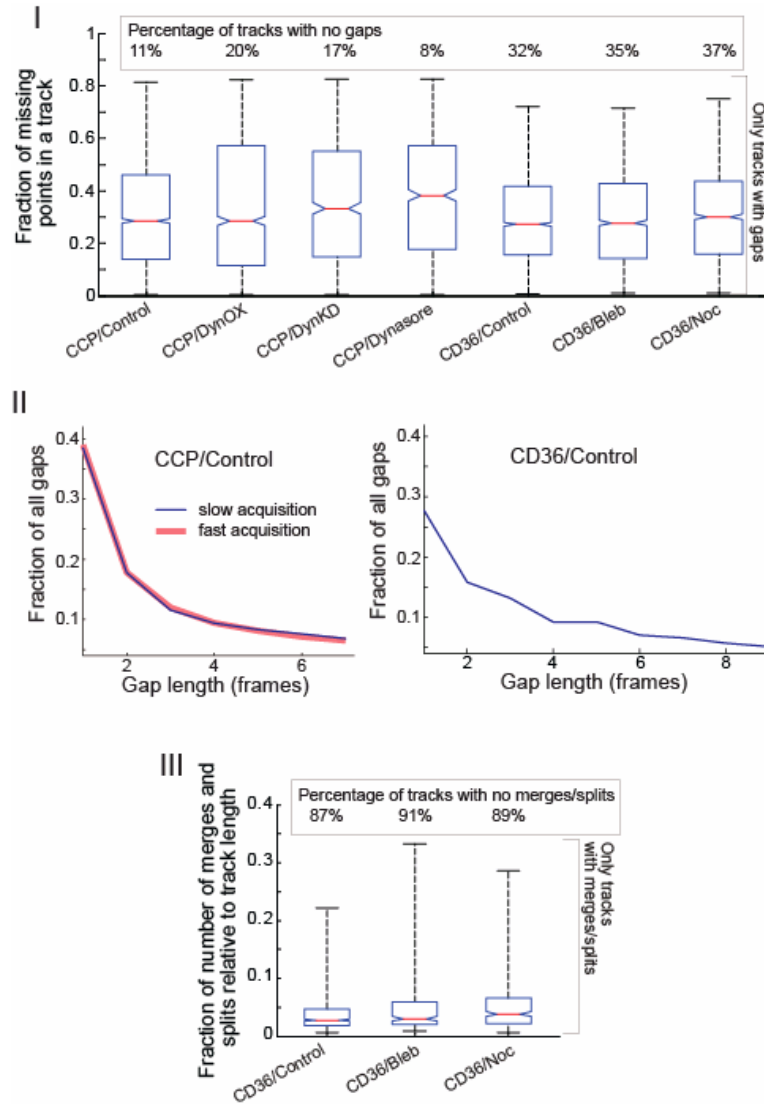
Gap closing, merging and splitting statistics for tracked experimental data. (**I**) Fraction of missing points in a track, relative to track length. The box plots indicate the 25th and 75th data percentiles (lower and upper lines of the "box", respectively), the median (middle line) and the extent of the data (whiskers). (**II**) Distribution of gap lengths for control CCPs and CD36. Longer gaps are less abundant than shorter gaps, indicating that the time window chosen for gap closing captured most gaps in the system. Furthermore, the distribution of gap lengths, in number of frames, is the same whether CCP movies were taken at a frame rate of 0.5 Hz (slow acquisition, data used in this study) or 2.5 Hz (fast acquisition, data used elsewhere). The time scale invariance of the gap length distribution provides further evidence that the gaps closed resulted from detection failure in the noisy images. In contrast, if the closed gaps were erroneous links between true track terminations and true track initiations, the gap length distribution would exhibit a characteristic time scale. (**III**) Fraction of the number of merges and splits in a track, relative to track length. Box plots as in (I). No merging and splitting was performed for CCPs. In (I) and (III), the distributions shown exclude tracks that do not include gaps or merges/splits, respectively. The percentages of tracks with no gaps (I) and with no merges/splits (III) are shown in the boxes above the distributions.

Particle density in movies of CCPs and CD36 receptors

Experiment	Ave. disp.		Ave. n. n. dist		Frac. n. n. dist. < 2 x ave. disp.	Ave. # pot. assign-ments per particle	Frac. particles with > 1 pot. assignment	Ave. # pot. assign-ments per segment	Frac. segments with > 1 pot. assignment
	pixels	μm	pixels	μm					
CCP/Control	0.91	0.06	12.94	0.87	0.0000	1.02	0.11	1.61	0.43
CCP/DynOX	0.63	0.04	13.78	0.92	0.0000	0.90	0.01	0.56	0.13
CCP/DynKD	0.70	0.05	13.48	0.90	0.0000	0.92	0.01	0.10	0.27
CCP/Dynasore	0.68	0.05	12.63	0.85	0.0000	0.90	0.01	1.17	0.36
CD36/Control	1.70	0.11	15.02	1.01	0.0003	0.84	0.04	0.80	0.19
CD36/Blebbistatin	1.44	0.10	16.30	1.09	0.0002	0.84	0.03	0.69	0.14
CD36/Nocodazole	1.75	0.12	19.36	1.30	0.0005	0.80	0.03	0.66	0.14

Parameters characterizing particle density in CCP movies (rows 1-4) and CD36 movies (rows 5-7). The columns show (**1,2**) the average particle displacement between consecutive frames, (**3,4**) the average nearest neighbor distance, (**5**) the fraction of particles with nearest neighbor distances smaller than twice their average frame-to-frame displacements, (**6**) the average number of potential assignments per particle in the frame-to-frame linking step, (**7**) the fraction of particles with more than one potential assignment in the frame-to-frame linking step, (**8**) the average number of potential assignments per track segment in the gap closing, merging and splitting step, and (**9**) the fraction of track segments with more than one potential assignment in the gap closing, merging and splitting step. DynOX: dynamin over-expression. DynKD: dynamin knockdown.

Detection method 1: Clathrin-coated pits

The objective of this detection algorithm was to extract from TIR-FM images the sub-pixel locations of CCPs of various sizes, ranging from just below the theoretical resolution limit (~220 nm) to several hundred nanometers. We implemented an algorithm based on the method of spot extraction via multiscale products of significance-thresholded wavelet coefficients[1]. The selection of significant particles in this method is size-invariant, enabling simultaneous identification of both small CCPs whose size is below the resolution limit and larger CCPs. Furthermore, since in this method the decision whether a particle is significant is made locally, the algorithm is robust against global intensity variations due to uneven TIR-FM illumination, variable unspecific background, or uneven distributions of CCPs.

We tested the efficiency of CCP detection as a function of signal-to-noise ratio (SNR) via a simulation where we generated an image of multiple isolated sub-resolution particles (pixel = 67 nm, Airy disc radius = 220 nm), added noise, and compared the subsequent detection results with the original particle positions. The SNR of an image was taken as the average SNR of its particles, where the SNR of a particle was defined as the ratio of its signal (i.e. maximum intensity of its image) above background intensity to
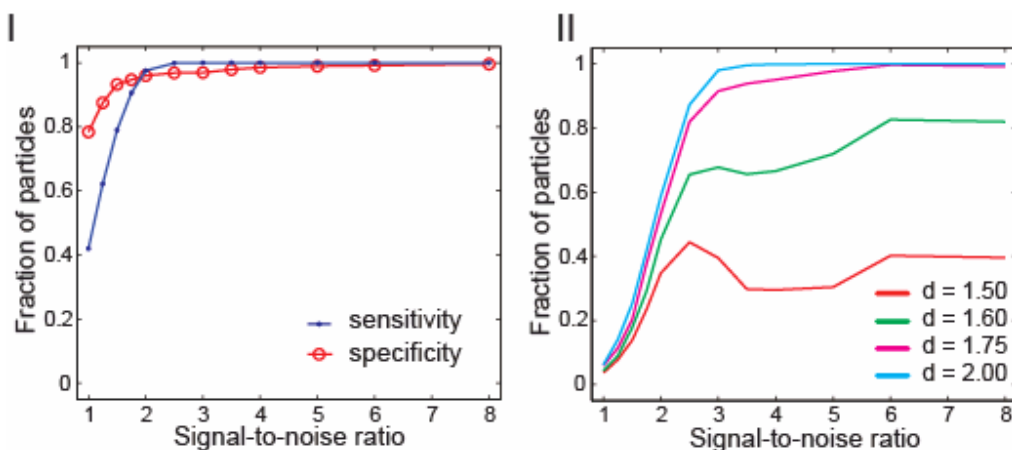
Figure A: CCP detection efficiency. (**I**) Sensitivity (number of true positives / ground truth) and specificity (number of true positives / (number of true and false positives)) for isolated sub-resolution particles. (**II**) Separation efficiency of particle pairs located at various distances = d × Airy disc radius.

the standard deviation of background intensity. At SNR values typical of low-intensity CCPs (2-5), the detection sensitivity of isolated particles was >90% (**Fig. AI**).

We also tested the efficiency of the detection algorithm in resolving particles that were very close to each other. For this purpose, we generated a distribution of diffraction-limited particle pairs at variable distances from each other, added noise, and determined the fraction of pairs that were successfully detected and resolved (**Fig. AII**). For SNR values typical of low-intensity CCPs (2-5), particles could be resolved with >50% efficiency when their separation exceeded 1.6 × the theoretical resolution limit (taken as the Airy disc radius[2]), corresponding to a distance of ~350nm.

Supplementary Note 2

Detection method 2: Single CD36 molecules

The goal of this detection algorithm was to provide the sub-pixel locations of sub-resolution particles in low SNR movies. Specifically, single-molecule CD36 movies had an SNR of 1.5-2. The SNR of a movie was calculated as the average SNR of its particles, where the SNR of a particle was defined as the ratio between its signal (i.e. maximum intensity of its image) above background intensity and the standard deviation of background intensity. To achieve sub-pixel localization, the algorithm fitted Gaussian kernels approximating the two-dimensional point spread function of the microscope around local intensity maxima, where both position and amplitude were free parameters in the fit. To detect significant local maxima to initialize the fit under low SNR conditions, the algorithm employed time averaged frames. This effectively decreased the noise-induced variation in the background intensity, enhancing the detection of true signals without increasing the number of false positives (**Fig. B** and **Supplementary Video 2**).
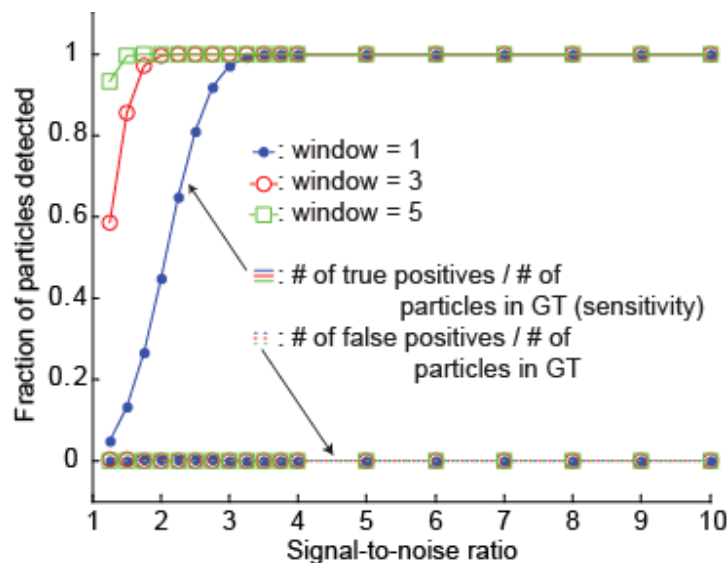
Figure B: Time averaging enhanced detection efficiency. Particles (Airy disc radius = 4.5 pixels, pixel = 67 nm) placed on a regular grid against backgrounds of increasing noise were detected and the detection results were compared to the ground truth. A time averaging window of one means no time averaging was performed. The same statistical test cutoffs were used in all cases.

In more detail, detection was achieved in three steps:

Step 1.   Time averaging: We employed time averaging to reduce the background intensity variation, thus enhancing the SNR, of CD36 movies. Assuming that background intensity was random over space and time, time averaging with a sliding window w would reduce the background intensity standard deviation by a factor of $\sqrt{w}$.

Time averaging is most effective when the particles to be detected do not move more than the Gaussian kernel within the frames of the sliding window. Otherwise, the particles in the averaged image get blurred, precluding the detection of one unique intensity maximum per particle. Thus, while for fixed particles the sliding window can be made as large as needed to reduce background intensity fluctuations, for moving particles in practice, the extent of particle motion between frames sets an upper limit on window size.

CD36 molecules moved on average ~1.5 pixels between frames. With this speed, the overlap between Gaussian kernels in consecutive frames was sufficiently large to not blur the time averaged images. However, between nonconsecutive frames, the displacement of kernels was too large to produce a unique local maximum per particle in the time averaged images. Thus, we limited time averaging to a window of three. This enhanced the sensitivity of local maxima detection in CD36 movies from ~40% to ~90% (**Fig. B**).

Step 2.   Local maxima detection in time-averaged images: A pixel was classified as a local maximum if its intensity was the highest in a 3×3 pixel area centered at that pixel. Local maxima were then tested against the variation in background intensity. Maxima whose intensity was above the background by more than 1.3×background standard deviations were classified as significant and thus retained (corresponding to a type-I error probability of 0.1 in the hypothesis test).

Step 3.   Sub-pixel localization: Gaussian kernels were fitted around every local maximum accepted in Step 2, following the approach described in Ref.[3]. While local maxima were detected in the time-averaged images, Gaussian kernel fitting was performed in the raw images in each frame. In brief, first one Gaussian kernel was fitted around a local maximum. Then it was attempted to fit a second kernel. If the residuals from fitting two kernels were statistically significantly smaller than the residuals from fitting one kernel, this was evidence for the existence of an additional particle that was missed by the initial search for local maxima. In this case, we retained the second kernel and then attempted to fit a third kernel. This procedure was repeated until there was not enough statistical evidence supporting the addition of a kernel.

Supplementary Note 3

Additional information for random motion cost matrix

In the section entitled "Tracking particles via spatially and temporally global assignments", we describe the costs for linking two particles (Eq. 3), for closing a gap between two track segments (Eq. 4) and for the merging and splitting of track segments (Eqs. 5 and 6), under the assumption of isotropic random particle motion. Here we complete the definition of this cost matrix by describing:

(1) The cutoffs used to prohibit physically nonsensical assignments. Any particle or track segment pairing with costs above the cutoffs was excluded a priori (*x*'s in **Fig. 1b,c**). In practice, this also reduced the computational complexity of the LAP.

(2) The alternative costs (*b*, *d*, *b*' and *d*' in **Fig. 1b,c**) that allowed particles and track segments to reject their potential assignments.

Given these definitions, the cost matrix was constructed in three steps: First, the cutoffs were estimated. Second, the costs of potential assignments were calculated. Third, the alternative costs were assigned.

In the end of the supplementary note, we also describe the auxiliary lower right block of the cost matrices in **Fig. 1b,c**.

Cutoffs:

Frame-to-frame linking:

- Cutoff 1: A search radius beyond which a link between two particles was deemed impossible. The search radius was determined for every particle on-the-fly from the already constructed tracks and from the local particle density:
  - o We defined a motion-based search radius R_i'. If particle i in frame t had been tracked for some frames already, its motion-based search radius was set to $R_i' = 3\sigma_i$, where σ_i was the standard deviation of its displacements from one frame to the next (**Fig. C**). If particle i was not linked to anything in the past, its motion-based search radius was calculated in the same way but using the average standard deviation of the displacements of all other particles.
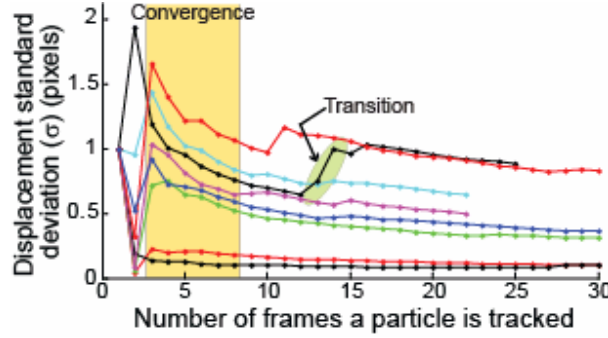
Figure C: Evolution of standard deviations (σ) over particle tracks, used for the definition of the search radius R'. Each particle has a distinct estimate of σ. Light orange rectangle: Domain of convergence of the estimate of σ (3-8 frames). Light green oval: Example of a particle undergoing a transition from slower to faster Brownian motion, which results in an adjustment of the standard deviation for this particle.

o   We defined an alternative search radius R''_i based on the local particle density. This was set to half the distance between particle i and its nearest neighbor. If particle i had been tracked for a few past frames, its nearest neighbor was determined not only using particle positions in frame t but also particle positions in past frames. This accounted for temporary particle disappearance, which could artificially increase a particle's nearest neighbor distance if the nearest neighbor happened to disappear. The number of past frames used was taken as the time window for gap closing (described below).

o   The final search radius R_i was then taken as $R_i = \max\left(R'_i, R''_i\right)$. With this definition, the search region of a particle located in a dense area was derived from its motion history and was likely to overlap with the search regions of other particles, requiring global solution of assignment conflicts by the LAP. On the other hand, if a particle was located in a sparse area, it could afford a search area larger than was required to cover the predicted range of motion. This is because in a sparse area it is less likely that another particle would appear (otherwise it would be a dense area).

o   This data-driven approach to setting search radii carries the risk that a few false initial assignments can cause inadvertent drifts toward either too slow or too fast motion. To prevent such drifts, the estimated search radii were bounded by empirical, user-defined lower and upper limits.

Gap closing, merging and splitting:
•   Cutoff 1: A time window over which gaps between track segment ends and starts may be closed. **Supplementary Note 4** discusses the effect of the time window on tracking efficiency.
•   Cutoff 2: A search radius per track segment start and end. Again, this parameter was defined in a data-driven procedure, accounting for motion and local particle density:
    o   An initial search radius $R'_I(\text{gap}) = 3\sigma_I \times f(\text{gap})$ was defined, where σ_I was the standard deviation of frame-to-frame displacements in track segment I. The function f(gap) scaled the search radius with $\sqrt{\text{gap}}$, with a potential plateau for

large gaps as described in **Supplementary Note 9**. This was the same for both segment start and segment end.

o Search radii $R''_{I,\text{start}}$ and $R''_{I,\text{end}}$ were defined based on local density as described above for particles. Importantly, the search radii of a track segment end and start could be different because of regional differences in particle density along a track segment.

o The final search radii were taken as $R_{I,\text{start}}(\text{gap}) = \max\left(R'_I(\text{gap}), R''_{I,\text{start}}\right)$ and $R_{I,\text{end}}(\text{gap}) = \max\left(R'_I(\text{gap}), R''_{I,\text{end}}\right)$.

o As with the particle search radii, $R_{I,\text{start}}$ and $R_{I,\text{end}}$ were bounded by empirical, user-defined limits.

- Cutoff 3: A lower and an upper limit on the ratio of intensities before and after merging and splitting. Potential merges and splits resulting in ratios outside this range were not allowed.

Alternative costs:

Frame-to-frame particle linking:

In live cell imaging, the number of particles detected in the frames of an image sequence is generally not constant. Particle number fluctuates because of the temporary disappearance of particles, the appearance of new particles and the disappearance of existing particles. Thus, when linking particles between two frames where the number of particles is not the same, it is unavoidable that some particles will have no counterpart in the other frame. Even if both frames contain the same number of particles, some particles are likely to have no counterpart. Thus, to be able to make a complete assignment between two frames, every particle in the source frame t should be allowed to link to nothing in the target frame $t + 1$, and every particle in the target frame should be allowed to get linked by nothing in the source frame.

These alternative costs for "no linking" (*b* and *d* in **Fig. 1b**) were inferred from the tracking information available up to the source frame t. They were taken as $1.05 \times$ the maximal cost of all previous links. Consequently, if all the potential assignments of a particle in frame t to particles in frame $t + 1$ had costs larger than this value, the particle would most likely link to nothing in frame $t + 1$. The same holds if a particle had potential links of lower costs, but it lost all the competitions against other particles. When a particle in frame t linked to nothing in frame $t + 1$, this resulted in the ending of a track segment. Symmetric considerations apply to particles in frame $t + 1$, in which case linking to nothing in frame t resulted in the starting of a track segment.

Gap closing, merging and splitting:

In this step, the algorithm attempts to distinguish track segment ends and starts resulting from temporary particle disappearance and particle merging and splitting from those that represent true track terminations and initiations. True terminations and initiations are respectively identified by segment ends and starts that get assigned to nothing. Thus, as in the particle linking case above, it is necessary to give every segment end and start the choice not to get assigned to any other segment.

In this step, we also introduce track segment middle points involved in potential merges with proximal segment ends and in potential splits with proximal segment starts

(**Fig. 1c**, upper middle block and middle left block). Since not every potential merge or split will have enough support in the data to get accepted, these segment middle points also require alternative costs for the rejection of potential merge and split events. Therefore, there were two categories of alternative costs in this step:

- For track segment ends and starts, the alternative cost (*b* and *d* in **Fig. 1c**) had to be comparable in magnitude to the costs of potential assignments, making the rejection of gap closing, merging and splitting an accessible alternative. At the same time, the alternative cost had to be at the higher end of the range of potential assignment costs, so that the algorithm did not fail to close gaps and capture merge and split events. We performed empirical tests of the sensitivity of tracking results to variations in the alternative cost. We found that in a range $80^{th} - 100^{th}$ percentile of all potential assignment costs the outcome of gap closing, merging and splitting varied negligibly (data not shown). We attribute this robustness to the fact that track initiations and terminations competed globally, in space and time, with all other potential assignments. Thus, the alternative cost was taken as the 90th percentile.
- For segment middle points introduced for merging and splitting, the alternative costs (*b'* and *d'* in **Fig. 1c**) were assigned on a case-by-case basis. Given the cost for the end/start of segment *I* merging with/splitting from a middle point of segment *J* (Eqs 5 and 6 in main text):

$$m_{IJ}, s_{IJ} = \begin{cases} \delta_{IJ}^2 \times \rho_{IJ}, & \rho_{IJ} > 1 \\ \delta_{IJ}^2 \times \rho_{IJ}^{-2}, & \rho_{IJ} < 1 \end{cases},$$

$$\rho_{IJ}\left(\text{merge at time } t\right) = \frac{A_J(t)}{A_I(t-1) + A_J(t-1)}, \quad \rho_{IJ}\left(\text{split at time } t\right) = \frac{A_J(t-1)}{A_I(t) + A_J(t)},$$

the alternative cost for the middle point of segment *J* refusing the merge/split was

$$b_J', d_J' = \begin{cases} \bar{\delta}_J^2 \times \rho_J', & \rho_J' > 1 \\ \bar{\delta}_J^2 \times \rho_J'^{-2}, & \rho_J' < 1 \end{cases},$$

where $\bar{\delta}_J$ denotes the average frame-to-frame displacement in segment *J* and

$$\rho_J'\left(\text{no merge at time } t\right) = \frac{A_J(t)}{A_J(t-1)}, \quad \rho_J'\left(\text{no split at time } t\right) = \frac{A_J(t-1)}{A_J(t)}.$$

The alternative cost compared the displacement δ_{IJ} needed for the merge or split to the average displacement $\bar{\delta}_J$ of track *J*, and the intensity ratio ρ_{IJ} resulting from merging or splitting to the intensity ratio ρ_J' that would result if the merge or split was refused. The alternative cost was taken as a characteristic of track *J* at the middle point of merging or splitting, rather than a characteristic of track *I*, in order to have a unique alternative cost in the cases where more than one ending or starting track segment attempted to merge with or split from track *J* at the same middle point.

For both cutoff and alternative cost calculations, initial guesses were employed whenever no information was available (e.g. when linking particles between the first and second frames). To minimize the effect of initial guesses on the final tracking result, i.e. to increase tracking robustness, we constructed the initial track segments in three rounds of frame-to-frame linking: forward in time, backward in time, and then forward again. Every round of linking employed the final estimates of particle displacement standard

deviations and alternative costs from the previous round, and thus the tracking results were eventually independent of user-defined initial guesses.

Auxiliary (lower right) block in the LAP cost matrix:

An assignment matrix $\hat{A}$ that is the solution to the LAP must have one and only one nonzero element per row and per column (Eq. 2). This applies also to rows and columns that are associated with alternative "no link" events ("no linking" rows and columns in **Fig. 1b**, "terminating" columns and "initiating" rows in **Fig. 1c**). If, for example, a particle is assigned to another particle in the frame-to-frame linking step, its associated "no linking" alternative event needs an auxiliary assignment to satisfy the LAP solution. While this assignment is meaningless in terms of tracking, it is a mathematical necessity of the LAP formalism.

To give these alternative events the possibility to get an assignment if not chosen by the particles and tracks, the lower right block of the cost matrix must be populated appropriately with selectable costs. Here we explain this procedure taking the frame-to-frame particle linking cost matrix as an illustrative example (**Fig. 1b**). For every potential assignment in the upper left block of the cost matrix, an auxiliary cost in the lower right block was needed. With this, whenever an actual assignment from the upper left block out-competed the assignment to non-linking alternative events in the upper right and lower left blocks, these non-linking events had the possibility for an auxiliary assignment through the lower right block. Given this condition, the structure of the lower right block was simply the transpose of the structure of the upper left block, thus preserving the sparseness of the assignment problem and minimizing the computational effort to search for auxiliary assignments. The auxiliary costs used in the lower right block were given the smallest values in the matrix, so that these costs did not influence the final solution.

Supplementary Note 4

Selection of optimal time window for gap closing

In the gap closing, merging and splitting step, the tracking algorithm attempts to link track segment ends to track segment starts in order to recover tracks interrupted by the temporary disappearance of particles. The time window over which gaps between segment ends and starts may be closed, i.e. the maximum gap length allowed, is a critical parameter in this step. A too restrictive window size will lead to broken trajectories, while a too large window size will tend to generate erroneous links between truly terminating particle tracks and the initiations of real new particle tracks in the vicinity. This raises the important questions of how to choose the time window for gap closing and whether, for a certain movie, this choice is sensitive, i.e. only one setting leads to reliable tracking, or robust, i.e. tracking delivers similar results within a range of time windows.

To answer this question, we tracked the particles in Simulation 1, combined with 20% detection misses (**Fig. 2**), using different time windows ranging from 1 (i.e. no gap closing) to 25 frames (i.e. the maximum allowed gap was 24 frames long). For each window setting, we compared the lifetime distribution resulting from tracking to the
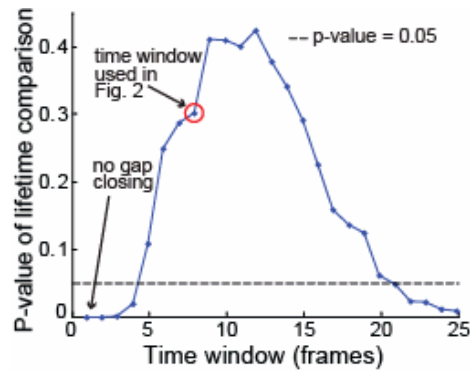
Figure D: Optimal time window for gap closing. Kolmogorov-Smirnov test p-value for comparing the GT lifetime distribution to the distribution reconstructed via tracking with various gap closing time windows.

simulated lifetime distribution (ground truth, GT) using the Kolmogorov-Smirnov test (**Fig. D**).

As expected, very small and very large time windows (e.g. 3 and 21 frames, respectively) resulted in strong disagreement between the reconstructed and GT lifetime distributions (p-values < 0.05). In between these values, there was a range of time windows (5-20 frames) which yielded lifetime distributions that were statistically indistinguishable from the GT (p-values > 0.05). Interestingly, time windows of 9-12 frames yielded better agreement than the time window used in **Fig. 2**, which was taken as 8 because the largest gap observed in the GT tracks was 7 frames long.

These results indicate that, while the time window for gap closing is a critical tracking parameter, there is a range of time windows, e.g. 8-14 frames for 20% detection misses, that yields equivalent tracking results. We attribute this robustness to the spatial and temporal globality of our tracking algorithm. As soon as parameter settings include true positive potential particle links, closed gaps and merge and split events, the chances are high that they will be selected as the best ensemble solution in a global competition against ensembles that also include false positives.

Supplementary Note 5

The imaged CD36 receptors are surface-bound

Our finding that a subset of CD36 receptors moved along linear tracks could suggest that some of the imaged fluorescent receptors were endocytosed and were thus being transported along microtubules inside the cell. To ensure that all the imaged CD36 receptors resided on the outer aspect of the plasma membrane of macrophages, we performed an acid wash experiment (**Fig. E**): First, short 20-frame movies (to avoid photobleaching) of CD36 molecules immuno-labeled with Fab-Cy3 were acquired (as described in Materials and Methods). The cells were then treated with acid buffer (200 mM acetic acid and 150 mM NaCl; pH 2.5) for 5 min and imaged again for 20 frames after placing the cells back in Hepes-buffered RPMI solution.

Acid wash is a standard method for confirming that antibodies are surface accessible. In particular, if the CD36 molecules had undergone endocytosis, the Cy3-Fab

fragments would have been internalized and would therefore not be affected by the acid wash. Conversely, if CD36 was on the membrane, the associated Cy3-Fab fragments would be outside the cell, and would thus get stripped by acid wash. The latter was found to be the case, with 85% of the fluorescent particles removed by the acid wash (**Fig. EI,II**), supporting the notion that the CD36 molecules studied resided in the plasma membrane.

The fraction of CD36 molecules remaining after acid washing (15%) was much smaller than the fraction of CD36 molecules undergoing linear motion in control cells (35%). Thus, even under the assumption that all of the CD36 molecules remaining after acid wash were indeed endocytosed, they could not account for all the CD36 molecules exhibiting linear motion. Therefore, at least half of the CD36 molecules moving along linear tracks had to be in the membrane. Furthermore, while CD36 receptors showed both linear and random motion before acid wash, with the characteristic radial arrangement of linear tracks (**Fig. EIII**), none of the remaining 15% of CD36 molecules after acid wash exhibited linear motion (**Fig. EIV**). Thus, provided that acid wash had little impact on intra-cellular molecular transport (i.e. it did not lead to the immobilization of internalized vesicles), these observations suggest that the subpopulation of CD36 molecules moving along linear tracks was surface-bound.

We further confirmed that the loss of fluorescence after acid wash was not due to direct effects of the acid medium on Cy3 fluorescence. To achieve this, we performed the same acid wash experiment on cells with fixed Fab fragments (using 8% paraformaldehyde for one hour before imaging and subsequent acid wash). In this case, most of the fluorescence remained, with only a 15% decrease in the number of fluorescent particles (**Fig. EV,VI**), likely reflecting a fraction of incompletely fixed antibodies. Hence, the observed reduction in particle number after acid wash during live cell experiments is related to the release of Cy3-Fab fragments bound to cell surface receptors, implying that the imaged CD36 receptors were in the plasma membrane.
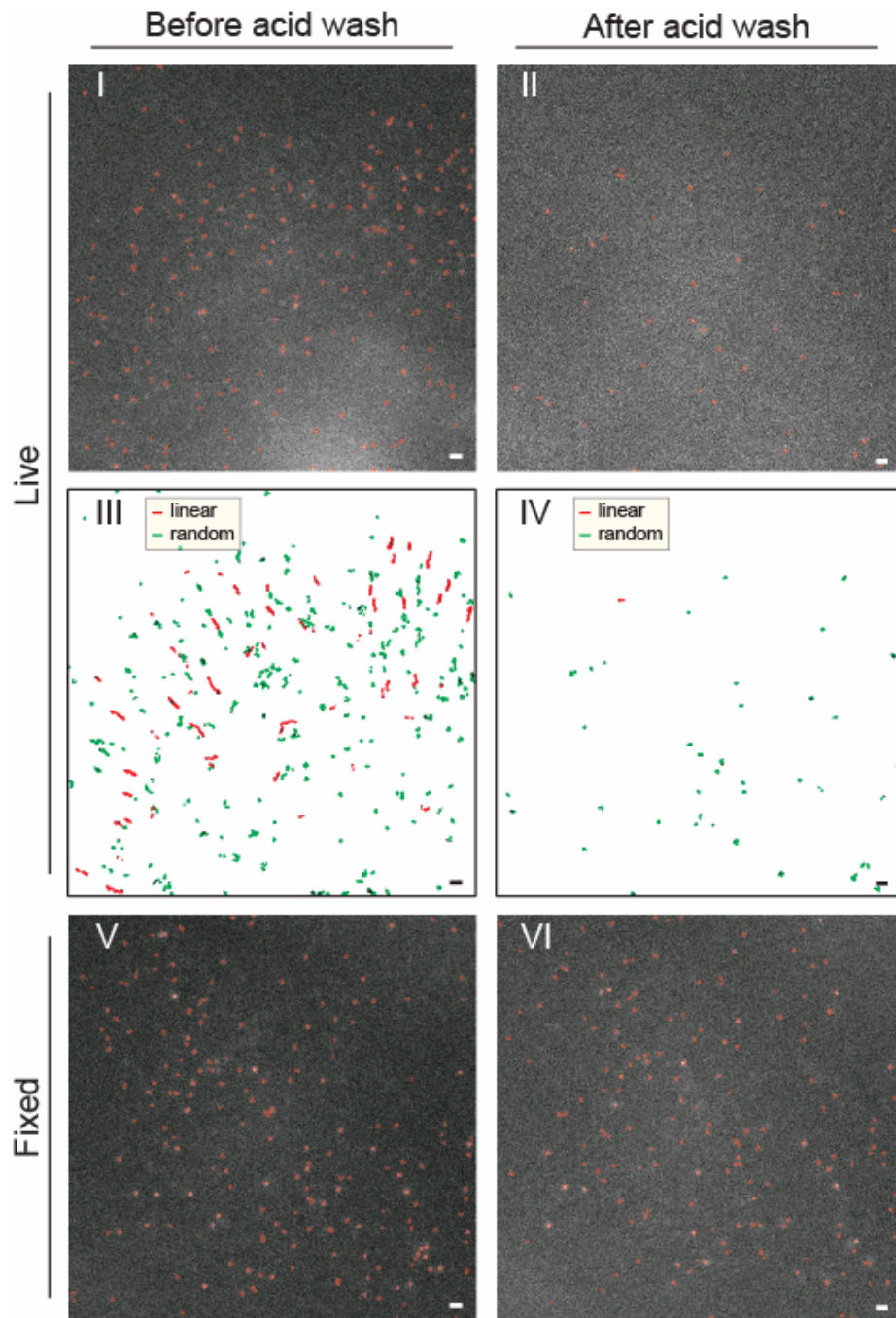
Figure E: CD36 molecules immuno-labeled by a primary Fab fragment and a Cy3-conjugated secondary Fab fragment reside in the plasma membrane of human macrophages. (**I,II**) Live-cell image and detection results before (**I**) and after (**II**) acid wash. (**III,IV**) CD36 tracks in a 20-frame live-cell movie (10 Hz frame rate) before (**III**) and after (**IV**) acid wash. (**V,VI**) Fixed-cell image and detection results before (**V**) and after (**VI**) acid wash.

Classification of CD36 trajectories as linear or random

For tracking as well as the analysis of the motion-conditional probabilities for merge and split events, we classified CD36 trajectories as linear or random based on the asymmetry in the scatter of particle positions along each trajectory, using a 2D implementation of the approach presented in Ref.[4]. This approach is built on the premise that the scatter of positions making a Brownian trajectory is symmetric, while the scatter of positions making a trajectory with preferred directionality is asymmetric.

The asymmetry S of a trajectory was defined as a function of the eigenvalues λ_1 and λ_2 of the variance-covariance matrix of the particle positions along the trajectory:

$$S = -\ln\left(1 - \frac{(\lambda_1 - \lambda_2)^2}{(\lambda_1 + \lambda_2)^2}\right).$$

For a perfectly symmetric trajectory, $\lambda_1 = \lambda_2$ and $S = 0$. For a perfectly asymmetric trajectory, $\lambda_2 = 0$ while $\lambda_1 \neq 0$, hence $S = \infty$. This method is especially suited for detecting asymmetries in short tracks, where a classical diffusion analysis would not be sensitive enough[4].

In order to reliably classify CD36 trajectories, we defined a significance threshold for S as a function of trajectory length. A length-dependent threshold was necessary because pure Brownian trajectories are not perfectly symmetric, and, the shorter a Brownian trajectory, the less symmetric it will appear. To determine the thresholds for trajectory lengths between 3 and 20 frames, we generated 2000 Brownian trajectories at every length, and calculated their asymmetry parameters. This generated a distribution of S values for every trajectory length. The threshold for significant asymmetry as a function of trajectory length was then taken as the 90th percentile of each distribution (**Fig. F**). We verified that these thresholds were independent of the Brownian motion diffusion coefficient by performing two separate simulations with different diffusion coefficients (D=1 and D=0.1 pixels^2/unit time) (**Fig. F**).
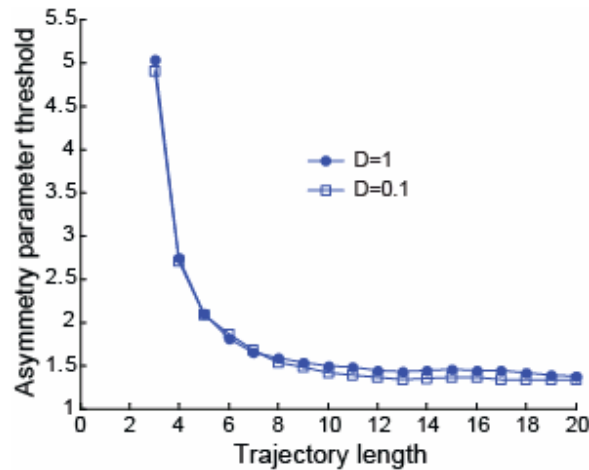

Figure F: Asymmetry parameter threshold for distinguishing linear trajectories from random trajectories. At every trajectory length, the threshold is the 90th percentile of the distribution of asymmetry parameter values of Brownian trajectories. The threshold reaches a plateau for trajectories longer than 10 frames.

For CD36 trajectory analysis, we used a length dependent threshold for trajectories 5-10 frames long (we did not classify or analyze trajectories shorter than 5 frames), and then used the plateau value for trajectories longer than 10 frames.

Supplementary Note 7

Linear motion + random motion cost matrix

For the tracking of CD36 receptors, we extended the cost functions of Eqs 3-6 to explicitly account for the linear motion of a sub-population of CD36 receptors. The linear motion + random motion cost matrix was constructed in the same three steps as the random motion cost matrix: First, the cutoffs were estimated. Second, the costs of potential assignments were calculated. Third, the alternative costs were assigned. In the linear motion + random motion cost matrix, not only were the cutoffs and alternative costs self-adaptive, but so were the decision on whether a track was linear and consequently its direction of motion. Therefore, to minimize the effect of initial guesses on the final tracking results, we constructed the initial track segments in three rounds of frame-to-frame linking, as in the random motion case.

In the following, we describe the potential assignment costs, the cutoffs, and the alternative costs used in the linear motion + random motion cost matrix.

Potential assignment costs:
Frame-to-frame linking:
The cost for potentially linking particle i in the source frame to particle j in the target frame was no longer δ_{ij}^2 (Eq. 3), but rather the square of the distance δ'_{ij} between particle j and the propagated position of particle i into the target frame. In every frame, each particle had three possible propagated positions: it could take a step forward along the preferred direction, it could take a step backward along the preferred direction, or it could switch to random motion in which case the propagated position was the same as the observed position. This propagation was achieved via three parallel extended Kalman filters[5] (forward linear, backward linear, random) that competed with each other, allowing particles to move back and forth along a preferred direction (linear motion), and to switch between linear and random motion at any time.

Gap closing, merging and splitting:
In the gap closing, merging and splitting step, we classified a track segment as linear or random based on the asymmetry in particle positions along it (**Supplementary Note 6**). The cost for closing a gap between two linear track segments was modified to be

$$g_{IJ} = \delta_{IJ}^2 \times \left(1 + \left(\sin \alpha_{IJ}\right)^2\right)$$

where α_{IJ} is the angle between the directions of motion of track segments I and J. The cost for merging and splitting of linear track segments was the same as for the random motion case (Eqs. 5, 6).

Cutoffs:

Frame-to-frame linking:

- Cutoff 1: A search radius per particle, defined as $R_i = \max\left(R_i', R_i''\right)$ for particle i. Like for randomly moving particles, this search radius was the maximum of a motion-based search radius (R_i') and a search radius accounting for the local particle density (R_i'') (**Supplementary Note 3**). The motion-derived search radius, $R_i' = 3\sigma_i'$, relied on the standard deviation σ_i' of the distances between the predicted positions and the actual positions of particle i. This difference reflected the random component of particle motion, rendering the definition of R_i' used here equivalent to the definition in the purely random case (**Supplementary Note 3**).

Gap closing, merging and splitting:

- Cutoff 1: A time window over which gaps between track segment ends and starts were allowed to be closed.
- Cutoff 2: A search radius per track segment start and per track segment end. The definition of these radii depended on whether a track segment was classified as random or linear, using the asymmetry in particle positions along the track segment[4] (**Supplementary Note 6**). For a random track segment, the search radii were calculated as for the random motion cost matrix (**Supplementary Note 3**). For a linear track segment, the search radii were calculated as follows:
  - Two motion-based search radii $R_{I,\text{long}}'\left(\text{gap}\right)$ and $R_{I,\text{short}}'\left(\text{gap}\right)$ were calculated for the directions parallel to and perpendicular to the preferred direction of motion. $R_{I,\text{short}}'\left(\text{gap}\right) = 3\sigma_I' \times f\left(\text{gap}\right)$, where σ_I' was calculated as above for particles and $f(\text{gap})$ scaled the search radius with $\sqrt{\text{gap}}$, with a potential plateau for large gaps (**Supplementary Note 9**). $R_{I,\text{long}}'\left(\text{gap}\right) = 3\Sigma_I \times f'\left(\text{gap}\right) + 3\sigma_I' \times f\left(\text{gap}\right)$, where Σ_I was the standard deviation of displacements along the preferred direction of motion, and $f(\text{gap})$ and $f'(\text{gap})$ scaled the search radius with $\sqrt{\text{gap}}$, with potential plateaus for large gaps (**Supplementary Note 9**). We defined the search radius along the preferred direction of motion based on a one-dimensional diffusion model rather than a directed motion model, in which case $3\Sigma_I\sqrt{\text{gap}}$ would be replaced by $v_I \times \text{gap}$ (v_I = the average frame-to-frame displacement along the preferred direction of motion). This is because, in the case of CD36, $\Sigma_I \approx v_I$, in which case $3\Sigma_I\sqrt{\text{gap}}$ is more inclusive than $v_I \times \text{gap}$ for gaps < 10 frames (which is our time window for gap closing). With this definition of search radius, the tracker was able to close gaps during which a particle either underwent directed motion or one-dimensional diffusive motion.
  - Density-based search radii $R_{I,\text{start}}''$ and $R_{I,\text{end}}''$ were calculated as described above (**Supplementary Note 3**).

- o Using $R''_{I,\text{start}}$ and $R''_{I,\text{end}}$, $R'_{I,\text{short}}(\text{gap})$ was expanded at the start and the end of candidate tracks, if possible. $R'_{I,\text{long}}(\text{gap})$ was not increased based on local density, since it reflected the steps along the preferred direction.
  - o A lower limit was used for the search radii parallel and perpendicular to the preferred direction of motion. An upper limit was also imposed on the search radius perpendicular to the preferred direction of motion.
- Cutoff 3: A lower and an upper limit on the ratio of intensities before and after merging or splitting.
- Cutoff 4: An upper limit on the angle between the directions of motion of two linear tracks.

Alternative costs:
Same definitions as for the random motion cost matrix (**Supplementary Note 3**).

Shortcomings of tracking CD36 receptors with a random motion model

Fig. G, Part 1, shows the results of tracking CD36 receptors with the modified cost functions. **Fig. G, Parts 2-4,** show the tracking results using the original random motion cost functions. In this case, without an explicit linear motion model, the tracking of CD36 molecules moving along linear tracks required an increase in the distance cutoff. However, since the distance cutoff was isotropic (**Supplementary Note 3**), enlarging it increased not only the number of potential assignments along the direction of motion, but also perpendicular to it. Furthermore, since the linking cost was also isotropic (Eq. 3), a link perpendicular to the preferred direction of motion could be equally likely as a link parallel to it. As a result, tracking CD36 receptors with a random motion model generated many erroneous cross-links between particles along separate, adjacent tracks. As the distance cutoff was increased, it became more feasible to capture steps along the preferred direction of motion, but at the same time the number of erroneous cross-links between separate tracks increased.
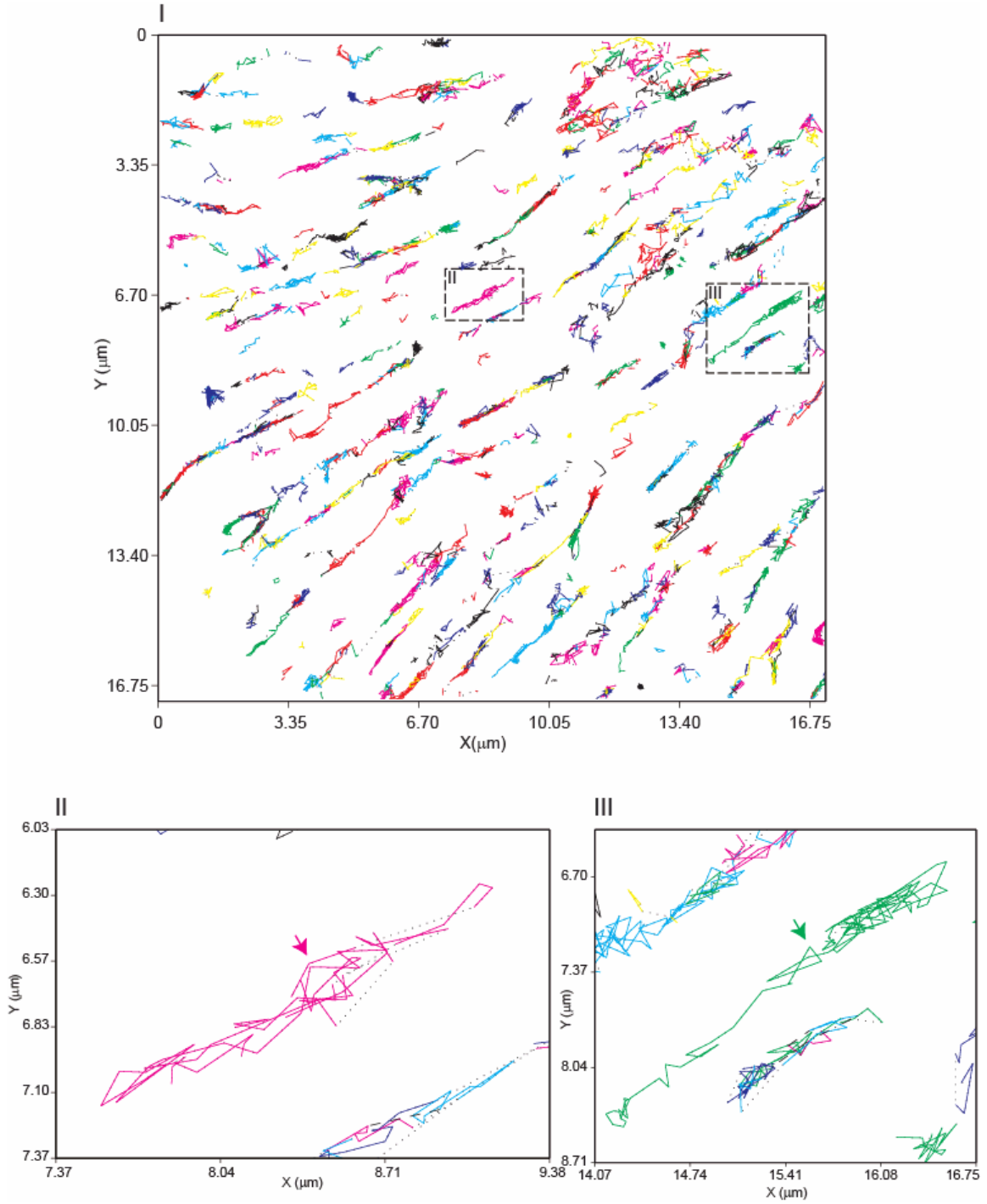
Figure G, Part 1: CD36 trajectories resulting from tracking with an explicit linear motion model. The different colors indicate different trajectories. (**I**) Whole field of view. (**II**) and (**III**) Zoom-in on two fully captured linear trajectories (pink arrow in (**II**), green arrow in (**III**)).
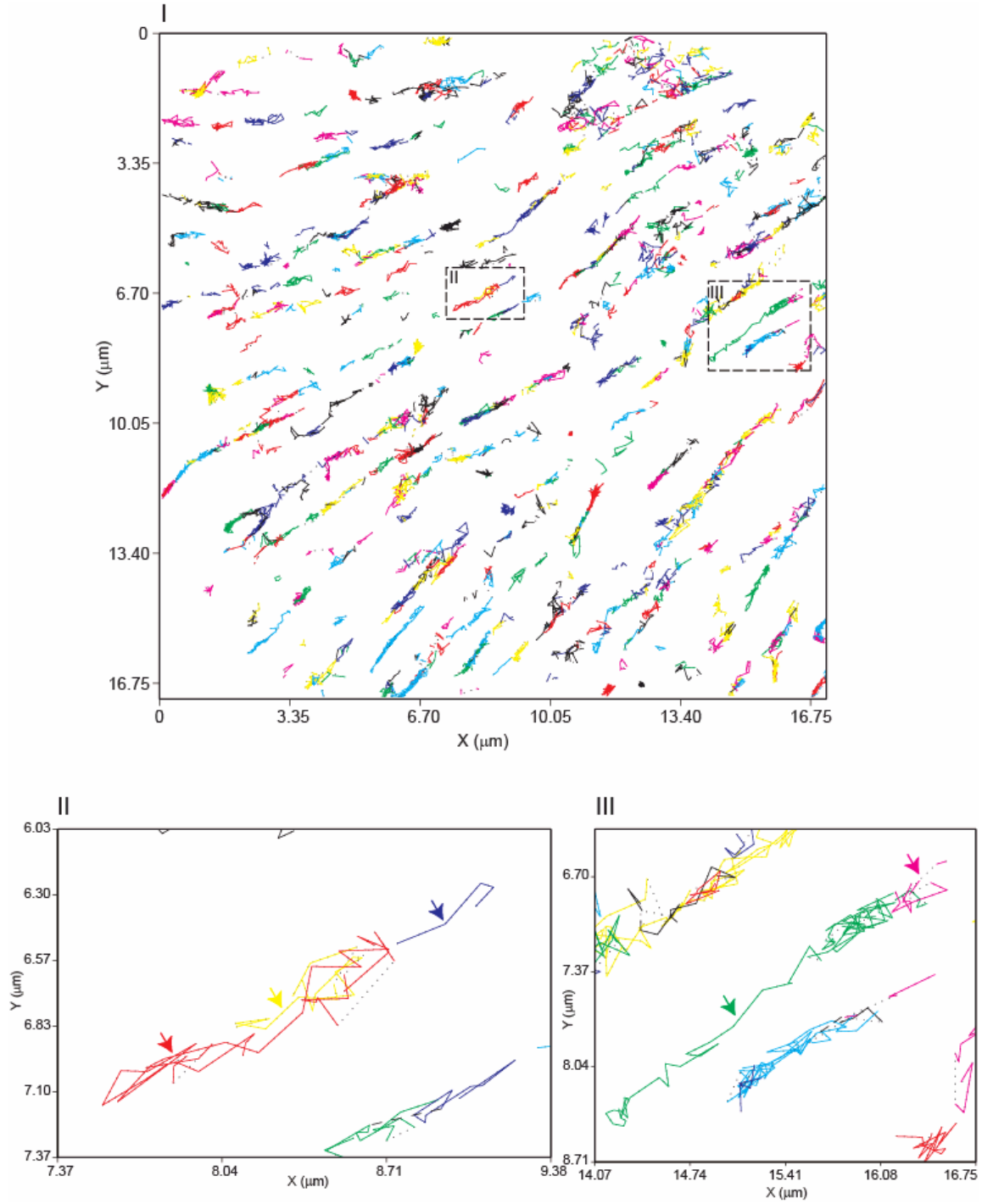
Figure G, Part 2: CD36 trajectories resulting from tracking without a linear motion model, using the same parameters and search radii as in Part 1. The different colors indicate different trajectories. (**I**), (**II**) and (**III**) are equivalent to (I), (II) and (III) in Part 1. Without a linear motion model, the trajectory in (**II**) got broken into three trajectories (blue, yellow and red arrows), while the trajectory in (**III**) got broken into two trajectories (pink and green arrows).
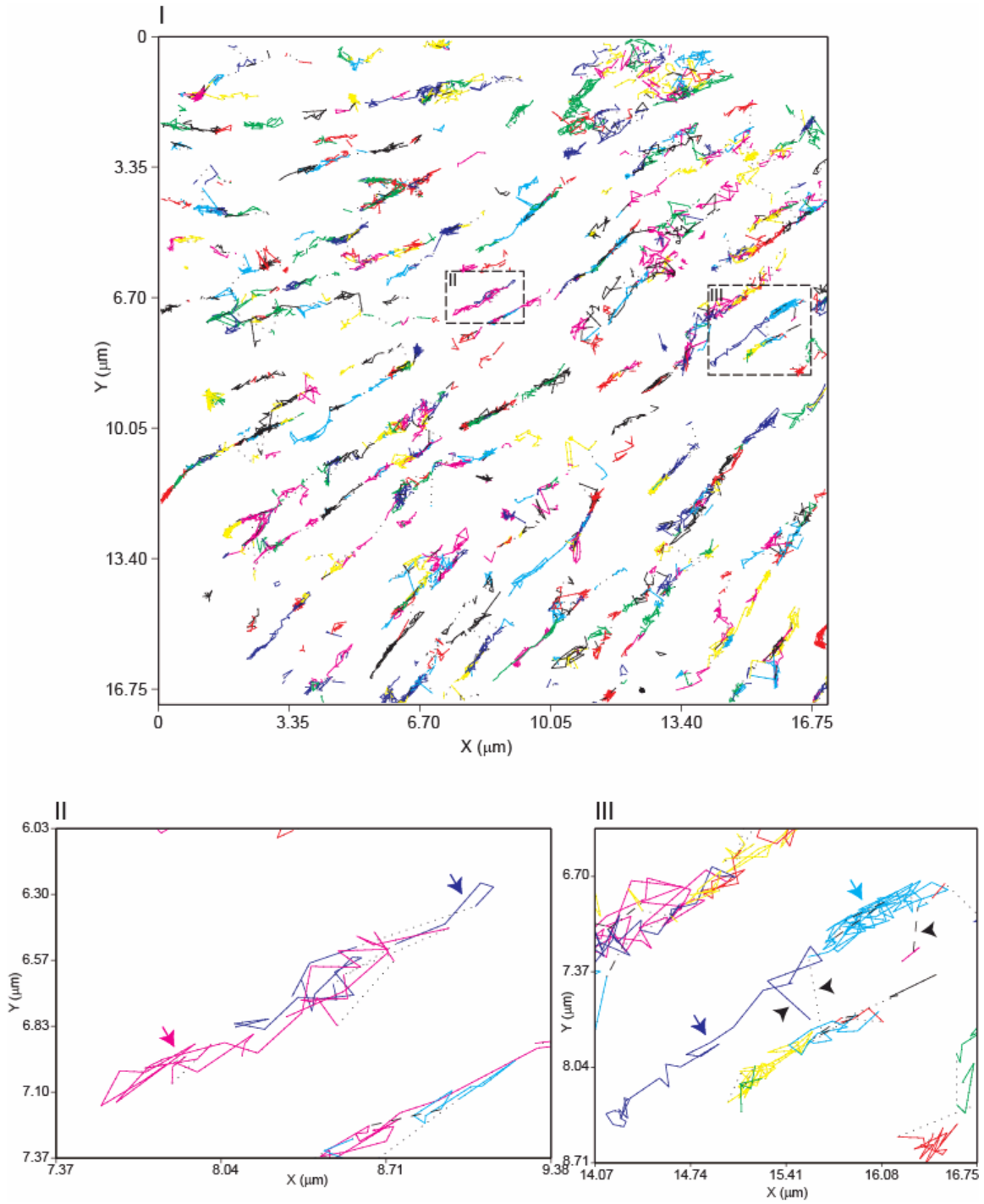
Figure G, Part 3: CD36 trajectories resulting from tracking without a linear motion model, allowing particles to link within twice the search radius used for tracking in Part 2. The different colors indicate different trajectories. (**I**), (**II**) and (**III**) are equivalent to (I), (II) and (III) in Parts 1 and 2. In this case, with larger search radii, the tracker produced erroneous cross-links between different trajectories (black arrowheads) while still not being able to completely link the trajectories in (**II**) (blue and pink arrows) and in (**III**) (cyan and blue arrows).
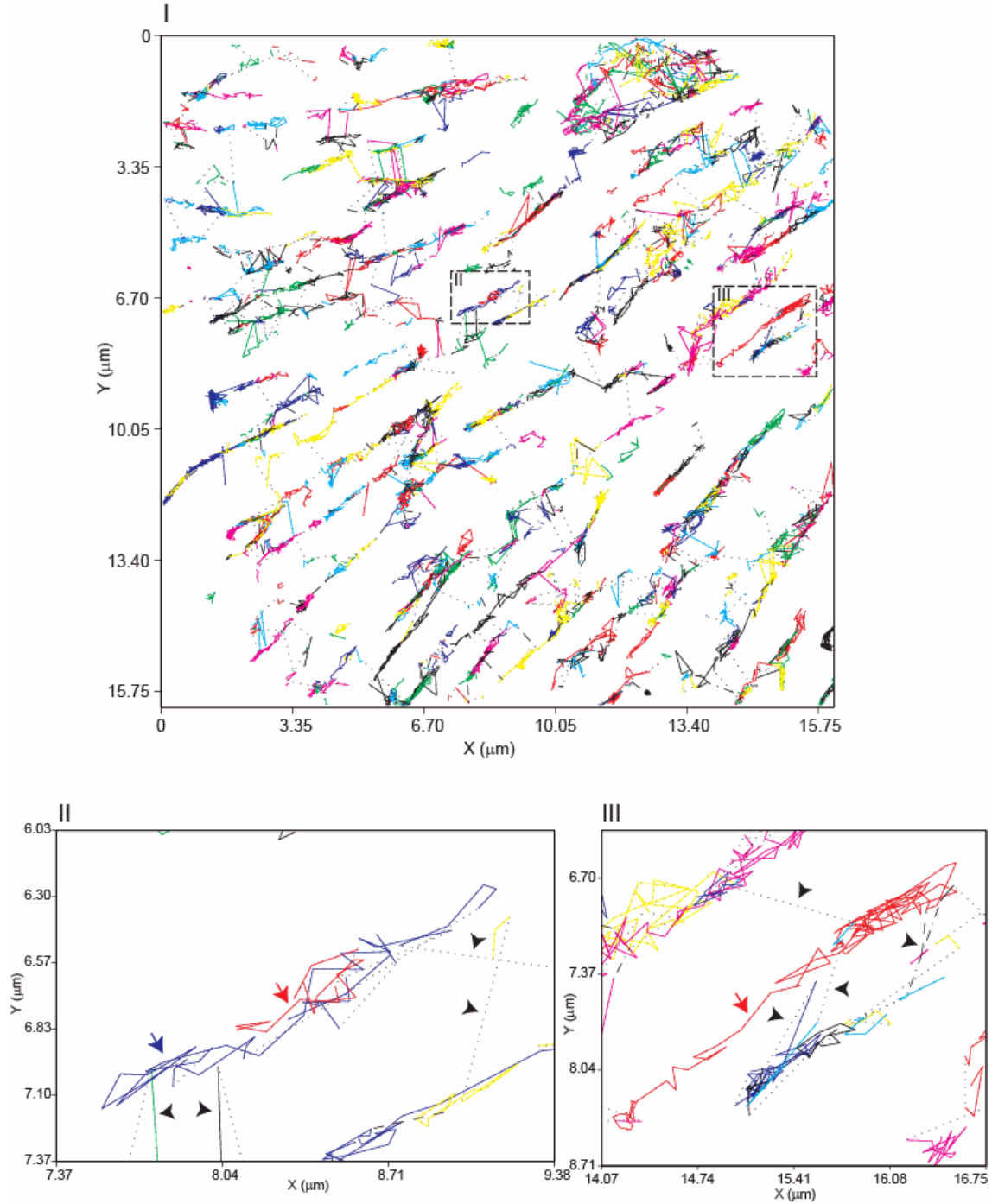
Figure G, Part 4: CD36 trajectories resulting from tracking without a linear motion model, giving particles three-times the search radius they had in Part 2. The different colors indicate different trajectories. (**I**), (**II**) and (**III**) are equivalent to (I), (II) and (III) in Parts 1, 2 and 3. While such a search radius was necessary to better capture the larger steps of particles in linear trajectories (the trajectory in (**III**) is now completely linked (one red arrow), the one in (**II**) is still not completely linked (red and blue arrows)), this search radius caused many erroneous cross-links between different trajectories (black arrowheads).

Merging and splitting probability analysis

The goal of this analysis was to estimate the conditional probabilities for merging and splitting events if a particle was undergoing linear motion, $p(ms\,|\,lin)$, and if a particle was undergoing random motion, $p(ms\,|\,rand)$. In the following, we focus the discussion on $p(ms\,|\,lin)$. The calculation of $p(ms\,|\,rand)$ was achieved analogously.

To estimate $p(ms\,|\,lin)$, we relied on Bayes' theorem on conditional probabilities[6]:

$$p(ms\,\&\,lin) = p(ms\,|\,lin) \times p(lin)$$
$$= p(lin\,|\,ms) \times p(ms)$$
(S1a)

$p(ms\,\&\,lin)$ is the probability of a particle merging/splitting AND undergoing linear motion, $p(ms\,|\,lin)$ is the conditional probability of a particle merging/splitting IF undergoing linear motion, $p(lin)$ is the probability of a particle undergoing linear motion regardless of its merge and split activity, $p(lin\,|\,ms)$ is the conditional probability of a particle undergoing linear motion IF merging/splitting and $p(ms)$ is the probability of a particle merging/splitting regardless of its motion type.

All terms on the right hand side of Eq. S1a, except the desired $p(ms\,|\,lin)$, could be calculated directly from the tracks. Therefore, we first calculated

$$p(ms\,\&\,lin) = p(lin\,|\,ms) \times p(ms),$$
(S1b)

from which we then calculated

$$p(ms\,|\,lin) = p(ms\,\&\,lin) / p(lin).$$
(S1c)

We obtained these probabilities in the following steps:

Step 1. Tracks were classified as either linear or random, using the asymmetry in the scatter of particle positions along each track[4].

Step 2. The probability of a particle merging/splitting $p(ms)$ was calculated as the ratio of the overall average number of merges/splits per frame, $\bar{N}_{ms}$, to the overall average number of particles in valid tracks per frame, $\bar{N}_p$:

$$p(ms) = \bar{N}_{ms} / \bar{N}_p.$$
(S2)

Valid tracks were tracks that lasted for at least 5 frames.

Step 3. The probability of a particle undergoing linear motion $p(lin)$ was calculated as the ratio of the average number of particles undergoing linear motion per frame, $\bar{N}_{p,lin}$, to the overall average number of particles per frame, $\bar{N}_p$:

$$p(lin) = \bar{N}_{p,lin} / \bar{N}_p.$$
(S3)

$\bar{N}_{p,lin}$ was obtained by finding in every frame the number of particles that belonged to linear tracks and then taking their average over all frames.

Step 4.   The conditional probability that a particle is undergoing linear motion IF it was merging/splitting, $p(lin\,|\,ms)$, was calculated as ratio of the average number of merges/splits per frame taking place in linear tracks, $\bar{N}_{ms,lin}$, to the overall average number of merges/splits per frame, $\bar{N}_{ms}$:

$$p(lin\,|\,ms) = \bar{N}_{ms,lin} / \bar{N}_{ms} \,. \tag{S4}$$

$\bar{N}_{ms,lin}$ was obtained by finding the number of merges/splits in every frame that belonged to linear tracks and then taking their average over all frames.

Step 5.   Substituting Eqs S2 and S4 in Eq. S1b yielded the probability of a particle merging/splitting AND undergoing linear motion, $p(ms\,\&\,lin)$.

Step 6.   Substituting the result from Step 5 and Eq. S3 in Eq. S1a yielded the conditional probability of interest, $p(ms\,|\,lin)$.

Validation of merging and splitting probability analysis for CD36

To ensure that our finding of a higher probability for merge and split events for linear tracks was not due to tracking bias and errors in the classification of linear vs. random tracks, we generated simulated particles and compared the reconstructed probability for merging and splitting with the ground truth probability.

Synthetic tracks were generated by Simulation #1 (**Fig. 2**), combined with 10% detection misses (reflecting an SNR ~2 with a time averaging window of 3 (**Fig. B**)). Half of the particle tracks in the simulation were modified by adding a linear component to the motion, i.e. by superimposing a one-dimensional diffusion component (with a diffusion coefficient of 1.5 pixels^2/frame) on top of two-dimensional diffusion. Thus, half the particles underwent linear motion (**Table A**), while the other half underwent random motion. Both types of particles had the same probability to merge and split (**Table A**).

We applied the merge and split analysis to both the ground truth and the recovered trajectories. In both cases, $p(lin)$ and the ratio of $p(ms\,|\,lin)$ to $p(ms\,|\,rand)$ were slightly underestimated, but were still close to the values used in the simulation. Importantly, these simulations show that the $p(ms\,|\,lin):p(ms\,|\,rand)$ ratio of ~2 that we find experimentally must reflect a truly higher frequency of merging and splitting events in linear tracks.

	$p(lin)$	$p(ms|lin)$	$p(ms|rand)$	$\dfrac{p(ms|lin)}{p(ms|rand)}$
Input parameters	0.50	0.035	0.035	1.00
Analysis of GT tracks	0.45	0.036	0.041	0.90
Analysis of tracking result	0.42	0.033	0.036	0.93

Table A: Merging and splitting parameters input to simulation (**1st row**) and the results of the merge and split analysis as applied to the ground truth tracks (**2nd row**) and the trajectories resulting from tracking (**3rd row**). The columns indicate (**1**) the probability of a particle undergoing linear motion, (**2**) the probability of a particle merging/splitting while undergoing linear motion, (**3**) the probability of a particle merging/splitting while undergoing random motion, and (**4**) the ratio between the two probabilities.

Supplementary Note 9

Tracking software details

In the following table, we describe the input variables for the tracking software described in this paper (available for download at lccb.scripps.edu, "download" hyperlink). In our tracking software, the actual cost functions themselves (highlighted in blue) are input variables, and thus can be exchanged for any other application-specific cost functions. The rest of the parameters described here are specific to the cost functions in our implementation (as described in the paper). Critical parameters that must be trained to the data are highlighted in orange. Flags (0/1) to choose tasks are highlighted in gray. Parameters that are not highlighted are not critical and can be used as supplied.

Input variable	Default	Notes
Cost functions		
Frame-to-frame linking	As described in the manuscript, for random and linear motion	Can be replaced by any user-defined functions that supply the proper input and output.
Gap closing, merging and splitting		Default cost functions can be self-adaptive or not, can follow linear and random motion, in 2D and in 3D.
Kalman filter functions		
Memory reservation	As described in manuscript, for random and linear motion	Required for default cost functions.
Filter initialization		Can be replaced by user-defined functions to suit different cost functions.
Gain calculation based on linking history		Can be skipped in case of non-adaptive cost functions.
General tracking parameters		
Gap closing time window	10 frames	Depends on SNR and fluorophore blinking. Critical if too small or too large. Robust if in proper range.
Flag for merging and splitting	1	1 to allow merging and splitting between track segments, 0 otherwise
Minimum track segment length used in the gap closing, merging and splitting step	2 frames	Since false positives during detection generally lead to tracks with short lifetimes, this parameter excludes very short tracks from participating in the gap closing, merging and splitting step. May need adjustment for very fast movies.
Cost function specific parameters		

Frame-to-frame linking		
Flag for linear motion	1	1 for position propagation prior to linking using a linear motion model (as described in the text), 0 otherwise (hence assuming random motion).
Search radius lower limit	2 pixels	To avoid drifts in self-adaptive estimation of search radius.
Search radius upper limit	5 pixels	Application-specific. To avoid drifts in self-adaptive estimation of search radius.
Standard deviation multiplication factor	3	Multiplication factor to calculate search radius from displacement standard deviation.
Flag for using local density in search radius estimation	1	1 to account for local particle density, in addition to the motion, in search radius calculation, 0 to use motion only.
Number of past frames used in nearest neighbor calculation	Gap closing time window	To account for possible neighbor disappearance when calculating nearest neighbor distance.
Cost function specific parameters		
Gap closing, merging and splitting		
Same parameters as for the frame-to-frame linking cost function	--	Same values as for the frame-to-frame linking cost function parameters. Can be adjusted for more sophisticated tracking.
Gap length at which f(gap) (in search radius definition; Notes 3 and 7) reaches its plateau	2 frames	$f\left(\text{gap}\right)=\begin{cases}\sqrt{\text{gap}}, \text{before transition point} \\ \left(\text{gap}\right)^{0.1}, \text{after transition point}\end{cases}$
Amplitude ratio lower and upper limits	[0.5 4]	To employ intensity information to determine whether a merge or a split is possible.
Minimum length for track segment analysis	5 frames	To classify a track segment as linear or random, or to keep unclassified if too short. To deduce track segment search radius from its history.
Standard deviation multiplication factor along preferred direction of motion	3	Multiplication factor to calculate search radius from the standard deviation of displacements along the preferred direction of motion.
Gap length at which f'(gap) (in definition of search radius parallel to preferred direction of	Gap closing time window	$f'\left(\text{gap}\right)=\begin{cases}\sqrt{\text{gap}}, \text{before transition point} \\ \left(\text{gap}\right)^{0.1}, \text{after transition point}\end{cases}$

motion; Note 7) reaches its plateau		
Maximum angle between the directions of motion of two linear track segments that are allowed to get linked	45 degrees	To employ directionality information to determine whether two linear track segments can get linked. This parameter assumes that the preferred direction of motion does not change over time.
Additional formulas embedded in cost functions		
For calculating alternative cost in the linking step	$1.05*max$(previous costs) when there are previous costs. 80th percentile of all potential costs in first frame (since there are no previous costs).	
For calculating alternative cost in the gap closing step	90th percentile of all potential assignment costs.	

Supplementary Methods

CCP labeling and imaging:

Cells and labeling:
BSC1 monkey kidney epithelial cells stably expressing rat brain clathrin light chain-EGFP were kindly provided by Dr. T. Kirchhausen, Harvard Medical School. BSC1 cells were plated at a density of 9×10^3 cells/cm² on 22x22 mm glass coverslips and preincubated for >30 min with imaging medium (DMEM, devoid of phenol red and containing 10 mM HEPES and 5% fetal bovine serum, pH 7.5) at 37°C. Coverslips were then mounted on a slide with two strips of double-sided tape spaced 1 cm apart. The volume between the strips of tape and the coverslip was filled with corresponding imaging medium and sealed with a 1:1:1 mixture of Vaseline, lanolin, and paraffin. For imaging, the slides were rapidly transferred to a prewarmed microscope stage (37°C; controlled with a custom-modified stage incubator; 20–20 Technologies). Four or five cells were imaged per coverslip.

Imaging:
Live cell TIR-FM imaging was performed using a 100X/1.45NA objective (Nikon) mounted on a Nikon TE2000U inverted microscope; the theoretical depth of the evanescent field was < 100 nm. Movies with a frame rate of 0.5 Hz were acquired using a 14-bit mode operated Hamamatsu Orca II-ERG camera.

CD36 labeling and imaging:

Macrophage isolation
Human blood samples from healthy volunteers were collected with heparin. Peripheral blood mononuclear cells were isolated by density-gradient centrifugation using Ficoll-

Paque Plus (Amersham) according to the manufacturer's instructions. Cells were resuspended (10^7 cells/ml) in RPMI 1640 with L-glutamine containing 10% heat-inactivated fetal calf serum (FCS; from Wisent) and seeded onto 18 mm glass coverslips (Fisher Scientific) at 5×10^5 cells/coverslip. After 1 h at 37°C, non-adherent cells were removed by sequential washes with Hanks buffered saline solution (HBSS). Adherent cells were incubated in RPMI 1640 with 10% FCS and 100 units/ml penicillin, 100 µg/ml streptomycin and 10 µg/ml polymyxin B (Invitrogen) for 7-14 days.

CD36 immunolabeling
Monoclonal antibodies to human CD36 (clone 131.1; mouse IgG1) were the generous gift of Dr. N. Tandon (Otsuka America Pharmaceutical, Inc., Maryland Research Laboratories, Rockville, MD). Monovalent Fab fragments were prepared using the ImmunoPure Fab Preparation Kit (Pierce), according to the manufacturer's instructions. To minimize non-specific binding during CD36 labeling the cells were blocked with 4% donkey serum for 10 min at 4°C, then incubated with anti-CD36 Fab fragments at a dilution of 1:3000 for 10 min at 4°C. After washing with HBSS, cells were incubated with Cy3-conjugated donkey anti-mouse Fab fragments (Jackson ImmunoResearch Laboratories) at a 1:3000 dilution for 10 min at 4°C.

Image acquisition
Live-cell imaging was performed using a Zeiss Axiovert 200 inverted epifluorescence microscope equipped with a 100x oil-immersion objective (NA 1.45) and a standard Cy3 filter set (Chroma Technology). Illumination was provided by an Exfo X-Cite 120 light source and a model 9100-13 deep-cooled EM-CCD camera (Hamamatsu) was used for recording. Image acquisition was controlled by the Volocity software (Improvision). To monitor CD36 mobility, images were acquired continuously at 10 frames per second for 10 s.

Reagents
Blebbistatin and Nocodazole were purchased from Calbiochem.

References

[1]   Olivo-Marin, J.-C., Extraction of spots in biological images using multiscale products. *Pattern Recogn* **35**, 1989 (2002).

[2]   Inoue, S. and Spring, K. R., *Video Microscopy: The Fundamentals*, 2nd ed. (Plenum, New York and London, 1997).

[3]   Thomann, D., Rines, D. R., Sorger, P. K., and Danuser, G., Automatic fluorescent tag detection in 3D with super- resolution: application to the analysis of chromosome movement. *J Microsc - Oxford* **208**, 49 (2002).

[4]   Huet, S. et al., Analysis of transient behavior in complex trajectories: application to secretory vesicle dynamics. *Biophys J* **91** (9), 3542 (2006).

[5]   Blackman, S.S. and Popoli, R., *Design and Analysis of Modern Tracking Systems*. (Artech House, Norwood, MA, 1999).

[6]   Papoulis, Athanasios, *Probability, Random Variables, and Stochastic Processes*, 3 ed. (McGraw-Hill, Inc., 1991).