

Multiple Object Tracking in Molecular Bioimaging by Rao-Blackwellized Marginal Particle Filtering

Essentially, all models are wrong, but some are useful.

— GEORGE E. P. BOX (1919 –)

Abstract — Time-lapse fluorescence microscopy imaging has rapidly evolved in the past decade and has opened new avenues for studying intracellular processes in vivo. Such studies generate vast amounts of noisy image data that cannot be analyzed efficiently and reliably by means of manual processing. Many popular tracking techniques exist but often fail to yield satisfactory results in the case of high object densities, high noise levels, and complex motion patterns. Probabilistic tracking algorithms, based on Bayesian estimation, have recently been shown to offer several improvements over classical approaches, by better integration of spatial and temporal information, and the possibility to more effectively incorporate prior knowledge about object dynamics and image formation. In this chapter, we extend our previous work in this area and propose an improved, fully automated particle filtering algorithm for the tracking of many subresolution objects in fluorescence microscopy image sequences. It involves a new track management procedure and allows the use of multiple dynamics models. The accuracy and reliability of the algorithm are further improved by applying marginalization concepts. Experiments on synthetic as well as real image data from three different biological applications clearly demonstrate the superiority of the algorithm compared to previous particle filtering solutions.

Based upon: I. Smal, E. Meijering, K. Draegestein, N. Galjart, I. Grigoriev, A. Akhmanova, M. E. van Royen, A. B. Houtsmuller, W. Niessen, “Multiple Object Tracking in Molecular Bioimaging by Rao-Blackwellized Marginal Particle Filtering”, *Medical Image Analysis*, vol. 12, no. 6, pp. 764–777, 2008.

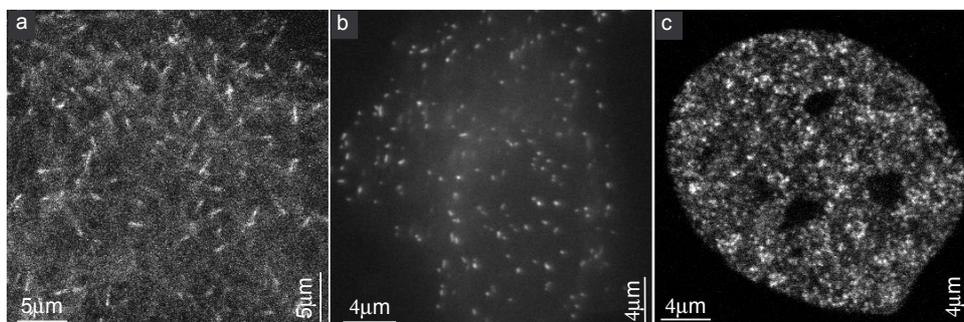


Figure 4.1. Sample images of microtubules (a), peroxisomes (b), and androgen receptors (c) (bright spots) labeled with green fluorescent protein (GFP) and imaged using confocal microscopy. The images are single frames from three 2D time-lapse studies, acquired under different experimental conditions. The quality of the images ranges from SNR ≈ 5 –6 (b) to ≈ 2 –3 (a,c).

4.1 Introduction

Advances in imaging technology for studying molecular processes in living cells continue to encourage biologists to conduct more and more challenging experiments and to collect large amounts of image data. Fluorescent labeling combined with time-lapse microscopy imaging enables visualizing the dynamic behavior of virtually any intracellular structure at high spatial and temporal resolution [53, 164, 180] (see Fig. 4.1 for example images). Quantitative analyses of this behavior requires the detection and tracking of large and time-varying numbers of nanoscale objects in the image sequences. Existing software tools (commercial and freeware) for this purpose are often not robust enough to yield satisfactory results when facing poor imaging conditions (very low-signal and high-noise levels are common in live cell imaging to minimize photodamage) and large numbers of objects with complex motion patterns (objects may interact or exhibit different modes of motion at different times). As a result, such analyses are still largely performed manually, by expert human observers. This is extremely labor intensive and very likely leads to user bias. Also, as only a part of the data can be analyzed this way, it may lead to the loss of important information. Therefore, the development of reliable automated algorithms, which allow the tracking of all individual objects moving along variable and unpredictable trajectories, constitutes an important first step in improving our understanding of the mechanisms controlling intracellular processes [52, 54, 95, 161].

The majority of approaches that have been proposed so far for tracking small objects in bioimaging applications consist of two stages. In the first stage, objects are detected separately in each frame of the image sequence, and in the subsequent second stage, an attempt is made to solve the interframe correspondence problem in linking detected objects between frames. Since the two stages are usually completely separated, without the possibility of feedback from linking to detection and vice versa, the tracking performance of such approaches is often suboptimal and ex-

tremely sensitive to failures in either stage. Moreover, as most of these approaches are designed to be generally applicable, they are often based on rudimentary detection algorithms (thresholding or template matching) and linking strategies (nearest neighbor or smooth motion). Recently, several popular approaches were quantitatively evaluated, and were found to break down below signal-to-noise ratios (SNRs) around 4–5 [26, 32], which are not uncommon in practice. More integrated, spatiotemporal segmentation approaches have also been proposed [17, 128], but current implementations of this idea have been demonstrated to work well only for single or very limited numbers of well-separated objects. More robust tracking approaches that can deal with larger numbers of objects have been developed for tracking of migrating cells using phase-contrast video microscopy [36, 83]. The robustness is achieved by exploiting the cell shape/appearance information, which cannot be utilized to the same extent for tracking of subresolution objects in fluorescence microscopy.

Most recently, probabilistic tracking approaches have been developed [52, 140, 145], which overcome the shortcomings of previous approaches by improved interaction between object detection and linking, and the possibility to more effectively incorporate prior knowledge about object dynamics and image formation. For example, for the tracking of growing microtubule plus-ends, whose dynamic behavior can be described accurately by a nearly constant velocity model, we have shown previously [141, 145] that a Bayesian estimation approach, in our case implemented by a sequential Monte Carlo (SMC) technique known as particle filtering (PF), makes better use of all available spatiotemporal information, yielding more accurate and more consistent tracking results (for more information about the success of the PF approach in other applications, and especially for tracking of multiple interacting objects, we refer to [39], [174], and [76]). However, that approach required a great deal of tailoring to the specific motion type to be analyzed, and was not able to directly deal with multiple motion types concurrently, nor with switching between them. It has also been shown [52] that an interacting multiple models (IMM) filter, which is capable of self-adapting to different motion types as well as to switching between them, can achieve more reliable tracking results than a Kalman filter (KF) using only one of the dynamics models. However, that approach did not optimally exploit all available spatiotemporal data, as the detection was implemented as a separate stage, completely decoupled from the linking stage.

In this chapter, we extend our previous work on the topic, and present an improved, fully automated algorithm for the tracking of many subresolution objects in time-lapse fluorescence microscopy images. Specifically, we take the successful particle filtering framework [140] as a starting point and propose five fundamental changes that make the algorithm more flexible, more robust, and more accurate. First, instead of using a single, dedicated dynamics model, multiple models are incorporated to be able to use the algorithm for different biological applications without the need for careful fine-tuning to each application. Second, a new detection scheme is integrated into the tracking framework, which is based on mean-shift clustering and performs better than the previously described classification approach. Third, a new likelihood evaluation strategy is proposed, which does not require the previously described “hierarchical searching” and reduces the computational cost. Fourth, we propose marginalization of the previously described filter, which increases the accuracy by reducing the vari-

ance of the track estimations. Finally, Rao-Blackwellization is applied to one of the state variables, which further improves the accuracy and reduces the computational cost, as it allows an analytical solution in the form of a Kalman filter. In addition to these methodological improvements, we extend our previous work by exploring two new biological applications, which could not be analyzed by our original algorithm [141] without careful tuning to each of these specific applications. By contrast, the algorithm proposed here can handle all of these applications without changing the parameter settings, as it naturally handles multiple and changing motion patterns.

The chapter is organized as follows. First, in Section 4.2, we recap the main ingredients of the particle filtering framework for multiple object tracking, and propose multiple dynamics models and a novel track management strategy. The subsequent two sections focus on the main novelties of the tracking approach compared to our previous work. In Section 4.3, we explain how multiple dynamics models can be conveniently incorporated into the particle filtering framework. Next, in Section 4.4, we show how to apply marginalization concepts to improve the performance of the framework. An overview of the algorithm and its parameters is given in Section 4.5. The results of experiments on synthetic as well as on real image data from three different biological applications are presented and discussed in Section 4.6. The evaluation includes a comparison with our previous algorithm [141] and with manual tracking, confirming the theoretically claimed improvements. Finally, in Section 4.7, we summarize the main findings of the present work.

4.2 Probabilistic Tracking Framework

The tracking approach proposed in this chapter is based on the principle of Bayesian estimation. In this section we first recap the Bayesian estimation framework and its implementation by means of particle filtering. Then we discuss two different ways of extending the framework to allow tracking of multiple objects. This is followed by a presentation of the dynamics and observation models that we propose for the biological imaging applications considered in this chapter. Finally we explain how we deal with track initialization, termination, and interaction within the framework.

4.2.1 Particle Filtering Approach

Bayesian estimation for tracking aims at inferring knowledge about the unobserved state \mathbf{x}_t of an object, which changes over time, using noisy measurements $\mathbf{z}_{1:t} \triangleq \{\mathbf{z}_1, \dots, \mathbf{z}_t\}$ up to time t . The evolution of the hidden state is assumed to be known and modeled as a Markov process of initial distribution $p(\mathbf{x}_0)$ and the transition prior $p(\mathbf{x}_t|\mathbf{x}_{t-1})$. The measurements $\mathbf{z}_{1:t}$, which are related to the state \mathbf{x}_t by the likelihood $p(\mathbf{z}_t|\mathbf{x}_t)$, are used to sequentially estimate the time evolving joint filtering distribution $p(\mathbf{x}_{0:t}|\mathbf{z}_{1:t})$ or the marginal filtering distribution $p(\mathbf{x}_t|\mathbf{z}_{1:t})$ and associated features, such as expectation. A recursive formula for the former is given by [39]

$$p(\mathbf{x}_{0:t}|\mathbf{z}_{1:t}) \propto p(\mathbf{z}_t|\mathbf{x}_t)p(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{0:t-1}|\mathbf{z}_{1:t-1}). \quad (4.1)$$

The distribution $p(\mathbf{x}_t|\mathbf{z}_{1:t})$ follows from (4.1) as

$$p(\mathbf{x}_t|\mathbf{z}_{1:t}) \propto p(\mathbf{z}_t|\mathbf{x}_t) \int p(\mathbf{x}_t|\mathbf{x}_{t-1}) p(\mathbf{x}_{t-1}|\mathbf{z}_{1:t-1}) d\mathbf{x}_{t-1}. \quad (4.2)$$

For most practical applications the optimal Bayesian solution defined by the recurrence relations (4.1) and (4.2) is analytically intractable. The exact solution can be obtained only for a restrictive set of cases [9], such as linear Gaussian modeling, in which case Kalman filtering can be applied. In order to solve the estimation problem in general, sequential Monte Carlo (SMC) methods, in particular particle filtering (PF) methods [9, 70], can be used as an efficient numerical approximation. The basic idea of such approximation is to represent the required posterior $p(\mathbf{x}_{0:t}|\mathbf{z}_{1:t})$ as a set of N_s random samples (particles) and associated weights $\{\mathbf{x}_{0:t}^{(i)}, w_t^{(i)}\}_{i=1}^{N_s}$:

$$p(\mathbf{x}_{0:t}|\mathbf{z}_{1:t}) \approx \sum_{i=1}^{N_s} w_t^{(i)} \delta(\mathbf{x}_{0:t} - \mathbf{x}_{0:t}^{(i)}), \quad (4.3)$$

where the weights are normalized, $\sum_{i=1}^{N_s} w_t^{(i)} = 1$, and $\delta(\cdot)$ is the Dirac delta function. The particle representation is updated in time according to (4.1) or (4.2) to obtain an approximation of the filtering distribution at successive time steps. The weights $w_t^{(i)}$ are obtained using sequential importance sampling (SIS) [40], which applies when auxiliary knowledge is available in the form of an importance density, $q(\mathbf{x}_{0:t}|\mathbf{z}_{1:t})$, describing which areas of the state space contain most information about the posterior. In order to calculate the weights recursively, the importance density is factorized as

$$q(\mathbf{x}_{0:t}|\mathbf{z}_{1:t}) = q(\mathbf{x}_t|\mathbf{x}_{0:t-1}, \mathbf{z}_{1:t}) q(\mathbf{x}_{0:t-1}|\mathbf{z}_{1:t-1}). \quad (4.4)$$

The particle representation of the posterior at time t is obtained by augmenting the set of existing particles $\mathbf{x}_{0:t-1}^{(i)}$, with the new state $\mathbf{x}_t^{(i)} \sim q(\mathbf{x}_t|\mathbf{x}_{0:t-1}^{(i)}, \mathbf{z}_{1:t})$, allowing the weights $w_t^{(i)}$ in (4.3) to be updated as [40]

$$\tilde{w}_t^{(i)} = \frac{p(\mathbf{x}_{0:t}^{(i)}|\mathbf{z}_{1:t})}{q(\mathbf{x}_{0:t}^{(i)}|\mathbf{z}_{1:t})} = \frac{p(\mathbf{z}_t|\mathbf{x}_t^{(i)}) p(\mathbf{x}_t^{(i)}|\mathbf{x}_{t-1}^{(i)})}{q(\mathbf{x}_t^{(i)}|\mathbf{x}_{0:t-1}^{(i)}, \mathbf{z}_{1:t})} w_{t-1}^{(i)}, \quad (4.5)$$

and normalized to

$$w_t^{(i)} = \frac{\tilde{w}_t^{(i)}}{\sum_{i=1}^{N_s} \tilde{w}_t^{(i)}}. \quad (4.6)$$

By using particle representations, statistical inferences such as expectation, maximum a posteriori (MAP), and minimum mean square error (MMSE) estimators, can be easily approximated [9].

4.2.2 Multiple Object Tracking

Tracking of multiple objects within the described PF framework can be done by extending each state vector $\mathbf{x}_t^{(i)}$ to include jointly the states of all objects at time t . This

approach is robust, but it drastically increases the dimensionality of the state space, leading to an exponential explosion of computational demands [39]. The alternative is to use an independent PF for each object. This approach is computationally cheap, but is prone to errors, especially during object interactions, where the track update for a given object may be contaminated with measurements from neighboring objects. Such cases require joint measurement-to-track association.

In our algorithm, we combine the best of both worlds: for objects that are far from other objects and do not interact at time t , the PFs are run independently, while for objects that come close to each other and do interact, we use joint sampling and updating of the weights, in combination with a reclustering procedure [145]. In order to accomplish this, the multimodal posterior distribution is represented as a mixture of individual non-parametric distributions, in the form

$$p(\mathbf{x}_t | \mathbf{z}_{1:t}) = \sum_{m=1}^M \pi_{m,t} p_m(\mathbf{x}_t | \mathbf{z}_{1:t}), \quad (4.7)$$

where M is the number of objects, and $\pi_{m,t}$ are normalized object weights, that is $\sum_{m=1}^M \pi_{m,t} = 1$. In this case, the particle representation of the filtering distribution consists of $N = MN_s$ particles, and is given by $\{\{\mathbf{x}_{m,t}^{(i)}, w_{m,t}^{(i)}\}_{i=1}^{N_s}\}_{m=1}^M$. This representation can be updated in the same fashion as the standard Bayesian sequential estimation, where the additional recursion for the mixture weights $\pi_{m,t}$ is given by [174]

$$\pi_{m,t} = \frac{\pi_{m,t-1} \sum_{i=1}^{N_s} \tilde{w}_{m,t}^{(i)}}{\sum_{n=1}^M \sum_{i=1}^{N_s} \pi_{n,t-1} \tilde{w}_{n,t}^{(i)}}. \quad (4.8)$$

4.2.3 Dynamics Models

As can be seen from (4.1) and (4.2), Bayesian tracking requires the specification of the transition prior, $p(\mathbf{x}_t | \mathbf{x}_{t-1})$, which models the dynamics of the objects to be tracked. This prior is application dependent and should be defined based on prior knowledge about the object motion patterns. In this chapter we are interested in the tracking of objects within the cytoplasm or the nucleus of biological cells. Eukaryotic cells contain numerous organelles and macromolecular structures. In most cases, the motion patterns of these objects are highly complex and difficult to describe by a single motion model.

In order to deal with different motion patterns, we consider two transition models, which together cover many of the patterns occurring in our applications. To this end, we define the state vector as

$$\mathbf{x}_t = (x_t, \dot{x}_t, y_t, \dot{y}_t, \sigma_{\max,t}, \sigma_{\min,t}, I_t)^T, \quad (4.9)$$

where $(\sigma_{\max,t}, \sigma_{\min,t})^T \triangleq \mathbf{s}_t$ is the object shape feature vector (discussed in more detail in Section 4.2.4), $(x_t, y_t)^T \triangleq \mathbf{r}_t$ is the position vector, $(\dot{x}_t, \dot{y}_t)^T \triangleq \mathbf{v}_t$ denotes velocity, and I_t intensity. Defining $\mathbf{y}_t = (x_t, \dot{x}_t, y_t, \dot{y}_t)^T$, and assuming that the changes in the position, intensity, and shape parameters are independent, we can factorize the state

evolution model as

$$p(\mathbf{x}_t|\mathbf{x}_{t-1}) = p_y(\mathbf{y}_t|\mathbf{y}_{t-1})p_s(\mathbf{s}_t|\mathbf{s}_{t-1})p_I(I_t|I_{t-1}). \quad (4.10)$$

For the combined position/velocity factor $p_y(\mathbf{y}_t|\mathbf{y}_{t-1})$, we define two models, denoted by $p_y(\mathbf{y}_t|\mathbf{y}_{t-1}, k)$, with $k \in \{1, 2\}$. For the first model ($k = 1$), which is suitable for tracking objects that exhibit motion patterns similar to random walk, the evolution of the state sequence is given by

$$\mathbf{r}_t = \mathbf{r}_{t-1} + T\boldsymbol{\xi}_t, \quad (4.11)$$

where T is the temporal sampling interval (that is, the time between any two successive time frames) and $\boldsymbol{\xi}_t$ is the process noise. The transition prior in this case is given by

$$p_y(\mathbf{y}_t|\mathbf{y}_{t-1}, k = 1) = \mathcal{N}(\mathbf{r}_t|\mathbf{r}_{t-1}, T^2 q_{1,1} \mathbf{I}), \quad (4.12)$$

where \mathbf{I} is the identity matrix and $\mathcal{N}(\cdot|\boldsymbol{\mu}, \boldsymbol{\Sigma})$ denotes the normal distribution with mean $\boldsymbol{\mu}$ and covariance matrix $\boldsymbol{\Sigma}$. For this type of motion, the velocity component \mathbf{v}_t does not influence the object position in (4.11), and for reasons discussed in Section 4.3, the $\mathbf{v}_t^{(i)}$ are uniformly sampled in the predefined interval $[V_{\min}, V_{\max}]$ at every time step.

For the second model ($k = 2$), which describes nearly constant velocity motion with small accelerations, we have [11]

$$\mathbf{y}_t = \mathbf{F}\mathbf{y}_{t-1} + \boldsymbol{\eta}_t, \quad (4.13)$$

where the process transition matrix \mathbf{F} and covariance matrix $\boldsymbol{\Sigma}$ of the process noise $\boldsymbol{\eta}_t$ are given by

$$\mathbf{F} = \begin{pmatrix} 1 & T & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & T \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad \text{and} \quad \boldsymbol{\Sigma} = \begin{pmatrix} \frac{T^3}{3} & \frac{T^2}{2} & 0 & 0 \\ \frac{T^2}{2} & T & 0 & 0 \\ 0 & 0 & \frac{T^3}{3} & \frac{T^2}{2} \\ 0 & 0 & \frac{T^2}{2} & T \end{pmatrix}. \quad (4.14)$$

The transition prior in this case is given by

$$p_y(\mathbf{y}_t|\mathbf{y}_{t-1}, k = 2) = \mathcal{N}(\mathbf{y}_t|\mathbf{y}_{t-1}, q_{1,2}\boldsymbol{\Sigma}). \quad (4.15)$$

In these two models, $q_{1,1}$ and $q_{1,2}$ are parameters, which need to be tuned experimentally to the applications.

The transition prior for the changes in object shape is defined using a Gaussian model, $p_s(\mathbf{s}_t|\mathbf{s}_{t-1}) = \mathcal{N}(\mathbf{s}_t|\mathbf{s}_{t-1}, Tq_2\mathbf{I})$, where q_2 represents the shape noise level. As for object intensity, in order to model the process of photobleaching, which in practice complicates the analysis of time-lapse fluorescence microscopy images [124, 148], a first-order Gauss-Markov process is used, with the transition prior $p_I(I_t|I_{t-1}) = \mathcal{N}(I_t|(1-\alpha)I_{t-1}, q_3T)$, where parameter q_3 accommodates fluctuations in object intensity, and $0 < \alpha < 1$ is estimated from the image data by fitting an exponential model [148] to the background intensity distribution as a function of time, as described previously in more detail [141].

4.2.4 Observation Model

The measurements in our applications are images of the intracellular objects obtained with optical microscopy imaging systems. Because the physical size of the objects (on the order of nanometers) is considerably smaller than the typical resolution of such systems (on the order of hundreds of nanometers), the intensity profiles of the objects in the images can be modeled using the point spread function (PSF) of the microscope. The described PF approach can accommodate any PSF that can be calculated pointwise. Commonly, a Gaussian approximation of the PSF is used [161] instead of theoretically more accurate models [190], because of its computational advantages and its accuracy, which is almost perfect for typical pinhole sizes (relative squared error less than 1%).

In order to model the image formation of the object profiles, one would have to use the convolution with the PSF for every state $\mathbf{x}_t^{(i)}$. To avoid this computational overload, we model the PSF together with object shape, using a Gaussian approximation. Spherical nanoscale objects are modeled using a rotationally symmetric Gaussian function. The intensity profiles of elongated objects are modeled by utilizing the velocity components from \mathbf{x}_t as parameters in the Gaussian. In this case, for an object of intensity I_t at position \mathbf{r}_t , the intensity contribution to pixel (i, j) is approximated as

$$h_t(i, j; \mathbf{x}_t) = a_t(i, j; \mathbf{r}_t, \mathbf{v}_t, \mathbf{s}_t)I_t + b_t(i, j), \quad (4.16)$$

where $b_t(i, j)$ is the background intensity and

$$a_t(i, j; \mathbf{r}_t, \mathbf{v}_t, \mathbf{s}_t) = \exp\left(-\frac{1}{2}\mathbf{m}_t^T \mathbf{R}^T \boldsymbol{\Sigma}_t^{-1} \mathbf{R} \mathbf{m}_t\right), \quad (4.17)$$

where $\mathbf{R} = \mathbf{R}(\phi_t)$ is a rotation matrix, $\boldsymbol{\Sigma}_t = \text{diag}[\sigma_{\max,t}^2, \sigma_{\min,t}^2]$, $\mathbf{m}_t = (i\Delta_x - x_t, j\Delta_y - y_t)^T$, and $\tan \phi_t = y_t/x_t$. Here, each pixel (i, j) is assumed to correspond to a rectangular area of dimensions $\Delta_x \times \Delta_y \text{ nm}^2$. The parameters $\sigma_{\max,t}$ and $\sigma_{\min,t}$ represent the amount of blurring and, at the same time, model object elongation along the direction of motion. For subresolution objects such as vesicles, $\sigma_{\min} = \sigma_{\max} \approx 100\text{nm}$, while for elongated structures such as microtubules, $\sigma_{\min} \approx 100\text{nm}$ and $\sigma_{\max} \approx 250\text{nm}$. The nonlinear dependence of the observations, modeled by (4.16), on the position, velocity, and shape parameters makes the described Bayesian inference analytically intractable. Contrary to the extended or unscented KFs [9], our PF-based approach is capable of preserving the multimodality in the posterior pdfs (see Section 4.2.2), which is exploited to resolve ambiguous object interaction events during tracking [141, 174].

The background level b_t in (4.16) is estimated by fitting the mixture of two Gaussian pdfs (for the background and the object intensity distribution, respectively) to the normalized intensity histogram in the current frame t [17]. Since the background is usually fairly uniform in our applications, we define the background level b_t as the mean of the background pdf. In the presence of background structures, which cause nonuniform background intensity, any local background estimation or subtraction algorithm can be used [117, 157].

4.2.5 Track Management

Automatic tracking of multiple objects also requires dealing with track initialization, initiation, termination, and object interaction scenarios. In other words, the number of objects, M in (4.7), is in fact time dependent, denoted as M_t . In order to initialize the tracker in the first frame, and also for the detection of newly appearing objects in subsequent frames, we use the h -dome transformation from (gray-scale) mathematical morphology [177]. The advantage of this transformation over other detection approaches is that all detected objects end up having the same maximum intensity in the resulting image. This is an important property for the detection scheme that we propose, as it ensures that the MC particles are properly distributed over the objects. The h -dome image $D_h(\mathcal{I}_t)$ of the h -domes of an image \mathcal{I}_t is given by

$$D_h(\mathcal{I}_t) = \mathcal{I}_t - \rho_{\mathcal{I}}(\mathcal{I}_t - h), \quad (4.18)$$

where the gray-scale reconstruction $\rho_{\mathcal{I}}(\mathcal{J}_t)$ of image \mathcal{I}_t from an image \mathcal{J}_t ($\mathcal{I}_t \geq \mathcal{J}_t$) is obtained by iterating gray-scale geodesic dilations of \mathcal{J}_t “under” \mathcal{I}_t until stability is reached [177]. Geometrically speaking, an h -dome D of image \mathcal{I}_t is a connected component of pixels such that every pixel p , neighboring D , satisfies $\mathcal{I}_t(p) < \min\{\mathcal{I}_t(q) | q \in D\}$ and $\max\{\mathcal{I}_t(q) | q \in D\} - \min\{\mathcal{I}_t(q) | q \in D\} < h$. The h -dome transformation extracts bright structures without requiring any size or shape criteria. In practice, a suitable value for h can be estimated from the image data, by relating it to the SNR. Specifically, in our implementation, we specify the minimum local SNR that an object is required to have in order for it to be included in the tracking, and h is set equal to the signal level corresponding to that SNR (see Section 4.6.2 for the definition of SNR).

Based on the h -dome transformation (4.18), we define the probability for the (real-valued) spatial position \mathbf{r}_t of objects at time t by the following transformation:

$$\tilde{p}(\mathbf{r}_t | \mathbf{z}_t) = (H_i H_n H_r D_h (G_\sigma * z_t))(\mathbf{r}_t), \quad (4.19)$$

where G_σ is the Gaussian kernel with scale σ (in practice we use $\sigma = 30$ – 50 nm), z_t is the image intensity value at time t , and H_i , H_n , H_r are operators acting sequentially on the h -dome image D_h . In other words, the image z_t is first convolved with the Gaussian kernel G_σ , and then h -dome transformed. The operator H_r creates a new image by raising the pixel values of the image D_h to the power $r > 0$. The operator H_n is used to normalize the intensity values of the newly created image, so they sum to 1 over the whole image domain. Finally, H_i creates a bilinear interpolation of the normalized image, which is then used to represent a proper pdf that can be evaluated at the real-valued position \mathbf{r}_t .

The function (4.19) is used as the sampling function, which generates MC samples in those regions where the probability of object existence is high. Having these samples in every frame, we apply the mean-shift algorithm [34], which clusters the particles into M_n classes. Next, we compare the number of particles in each class to a threshold, and if the number of particles in any class is greater than N_b , a new object is initiated for that class, where N_b is the expected number of particles in the vicinity of the object if the MC sampling were done uniformly over the image. Here, “vicinity” is defined

as a disk with radius $3\sigma_{\max, t=0}$, centered at the expected object position. In our experiments we sample $M_t N_s$ particles during this detection step. Thus, the threshold N_b is computed from the fact that for uniform sampling, the ratio between the disk area and the image area is equal to $N_b/(MN_s)$. During tracking, the detection is done in parallel, and detected objects are compared with the M_t existing ones. The M_b newly detected (born) objects are initiated with mixture weights $\pi_{m,t} = (M_t + M_b)^{-1}$, for $m = \{M_t + 1, \dots, M_t + M_b\}$, after which all $M_t + M_b$ mixture weights $\pi_{m,t}$ are renormalized.

Track termination is based on the analysis of the unnormalized weights $\tilde{w}_t^{(i)}$ in (4.5), using the likelihood ratio test [16]. Specifically, we check whether the average of the unnormalized likelihood values $p(\mathbf{z}_t | \mathbf{x}_t^{(i)})$ of all particles corresponding to a given object dropped below the level π_d of the likelihood (defined by a χ^2 -distribution, see Section 4.4.3), which corresponds to having no measurements from the object but only from the background. In the case of object interactions, when several of them approach each other, the particle representations of the posterior probabilities may become too diffuse, as measurements from other objects may contaminate the track updates. In order to deal with such cases, we employ a reclustering procedure [174]. This can be described by a function F , which maps the current representation into a new one, $(\{\mathbf{x}_t^{(i)}\}, \pi_{m,t}, M_t) \mapsto (\{\mathbf{x}'_t^{(i)}\}, \pi'_{m,t}, M'_t)$. In our algorithm, the mapping function is implemented as K -means clustering.

4.3 Incorporating Multiple Dynamics

The objects of interest in our applications exhibit quite different and complicated motion patterns that cannot be accommodated by the transition prior $p(\mathbf{x}_t | \mathbf{x}_{t-1})$ in a simple form. For accurate estimation and robust tracking, it is better to model each of the (sub)patterns by a separate transition prior, as described in Section 4.2.3. There are, however, no straightforward solutions to incorporating multiple dynamics models into the PF framework. In order to deal with different motion patterns, we propose to use jump Markov systems (JMS), where the state-space description allows for system parameter changes over time according to a Markov chain [41]. The frequently used interacting multiple model (IMM) filter is an example of a JMS in the case of linear Gaussian models [11]. In our case, however, we use the more general formulation suitable for our needs, assuming that the state prior $p(\mathbf{x}_t | \mathbf{x}_{t-1}, k_t)$ switches between K types of motion patterns, depending on the value of the parameter k_t .

The filtering distribution $p(\mathbf{x}_t, k_t | \mathbf{z}_{1:t})$ can be factorized as

$$p(\mathbf{x}_t, k_t | \mathbf{z}_{1:t}) = P(k_t | \mathbf{z}_{1:t}) p(\mathbf{x}_t | k_t, \mathbf{z}_{1:t}), \quad (4.20)$$

where $P(\cdot | \cdot)$ denotes the conditional probability mass function (pmf). The two factors in (4.20) are updated recursively in a so-called mixing stage and a mode-conditioned filtering stage. The mixing stage gives the predicted density $p(\mathbf{x}_{t-1}, k_t | \mathbf{z}_{1:t-1})$ for the modal state k_t as

$$p(\mathbf{x}_{t-1}, k_t | \mathbf{z}_{1:t-1}) = p(\mathbf{x}_{t-1} | k_t, \mathbf{z}_{1:t-1}) P(k_t | \mathbf{z}_{1:t-1}), \quad (4.21)$$

where

$$P(k_t|\mathbf{z}_{1:t-1}) = \sum_{k_{t-1}=1}^K P(k_t|k_{t-1})P(k_{t-1}|\mathbf{z}_{1:t-1}), \quad (4.22)$$

and

$$p(\mathbf{x}_{t-1}|k_t, \mathbf{z}_{1:t-1}) = \sum_{k_{t-1}=1}^K \frac{P(k_t|k_{t-1})P(k_{t-1}|\mathbf{z}_{1:t-1})}{P(k_t|\mathbf{z}_{1:t-1})} p(\mathbf{x}_{t-1}|k_{t-1}, \mathbf{z}_{1:t-1}). \quad (4.23)$$

In these equations, the probability of switching between models, $P(k_t|k_{t-1})$, is defined by a finite-state Markov chain, with transition matrix $\Pi = (p_{ij})$, where $p_{ij} = P(k_t = j|k_{t-1} = i)$. In the mode-conditioned filtering stage, we compute the second factor in (4.20) as

$$p(\mathbf{x}_t|k_t, \mathbf{z}_{1:t}) \propto p(\mathbf{z}_t|\mathbf{x}_t)p(\mathbf{x}_t|k_t, \mathbf{z}_{1:t-1}), \quad (4.24)$$

where the predicted density is given by

$$p(\mathbf{x}_t|k_t, \mathbf{z}_{1:t-1}) \propto \int p(\mathbf{x}_t|\mathbf{x}_{t-1}, k_t)p(\mathbf{x}_{t-1}|k_t, \mathbf{z}_{1:t-1})d\mathbf{x}_{t-1}. \quad (4.25)$$

The posterior mode probabilities are calculated as

$$P(k_t|\mathbf{z}_{1:t}) \propto P(k_t|\mathbf{z}_{1:t-1}) \int p(\mathbf{z}_t|\mathbf{x}_t)p(\mathbf{x}_t|k_t, \mathbf{z}_{1:t-1})d\mathbf{x}_t. \quad (4.26)$$

An efficient implementation of the described JMS approach was presented by [41] for radar tracking, involving dynamics models (constant velocity and circular turn) specific for that application.

For the initialization, the prior probabilities in the first time step are set to $P(k_0 = 1) = P(k_0 = 2) = 0.5$, which reflects the fact that we have no preference for any of the two models before the tracking commences. The velocity $\mathbf{v}_{t=0}^{(i)}$ for both models is sampled uniformly in the interval $[V_{\min}, V_{\max}]$. For model $k = 1$, this sampling is repeated for all frames during tracking (as explained in Section 4.2.3). This is necessary because if the dominant model is $k = 1$, the majority of the samples for $k = 2$ will be taken from model $k = 1$ after the mixing stage, but in order to “catch” the fast motion in such situations, the samples for the second model need higher velocity than the small displacements per time interval T during the random walk, so that they are propagated farther in space and match possible jumps.

4.4 Applying Marginalization Concepts

In the previous sections we have presented the general PF framework and the specific choices that we have made to tailor this framework to the problem of detecting and tracking multiple nanoscale objects exhibiting complex dynamics in biological imaging applications. Here we propose to further improve the framework by marginalization of the filtering distribution, data-dependent importance sampling, and Rao-Blackwellization. In the sequel, when we speak of the standard PF approach, we mean the described algorithm not including these improvements, and using only a single dynamics model.

4.4.1 Filtering Distribution Marginalization

In the standard PF approach, each particle $\mathbf{x}_t^{(i)}$ at time t , which augments the state path $\mathbf{x}_{0:t-1}^{(i)}$, is a draw from the joint space $p(\mathbf{x}_{0:t}|\mathbf{z}_{1:t})$, sampled sequentially. At each time step, the dimension of the sampled paths is increased by the dimension of the state space, n_x , quickly resulting in a very high-dimensional space. Because of the sequential nature of the algorithm, the variance of the importance weights can only increase (stochastically) over time [40], leading to most paths having vanishingly small probability [9, 39, 70].

One way to reduce this degeneracy effect is to apply marginal particle filtering (MPF), where the filtering is performed directly on the marginal distribution $p(\mathbf{x}_t|\mathbf{z}_{1:t})$, defined by (4.2), instead of on the joint state [77]. Having a representation of $p(\mathbf{x}_t|\mathbf{z}_{1:t})$ in the form of (4.3), we can approximate the integral in (4.2) as the weighted kernel estimate $\sum_{j=1}^{N_s} w_{t-1}^{(j)} p(\mathbf{x}_t|\mathbf{x}_{t-1}^{(j)})$. The importance weights are now on the marginal space:

$$w_t^{(i)} \propto \frac{p(\mathbf{x}_t^{(i)}|\mathbf{z}_{1:t})}{q(\mathbf{x}_t^{(i)}|\mathbf{z}_{1:t})} = \frac{p(\mathbf{z}_t|\mathbf{x}_t^{(i)}) \sum_{j=1}^{N_s} w_{t-1}^{(j)} p(\mathbf{x}_t^{(i)}|\mathbf{x}_{t-1}^{(j)})}{\sum_{j=1}^{N_s} w_{t-1}^{(j)} q(\mathbf{x}_t^{(i)}|\mathbf{x}_{t-1}^{(j)}, \mathbf{z}_t)}. \quad (4.27)$$

This MPF approach is potentially more robust against deviations in object dynamics compared to the specified models, and the variance of the importance weights for MPF is also lower than for standard PF [77]. However, these advantages exist only when the importance sampling function is dependent on the image data. By using only the transition prior $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ in the importance sampling, as commonly done in practice [9, 40, 70], the MPF simplifies to the standard PF, and all the benefits are lost.

4.4.2 Data-Dependent Sampling

In order to efficiently create MC samples at any time t , and to exploit the benefits of applying MPF, we propose to use a data-dependent importance sampling function. Specifically, we use a mixture of the state prior and a data-dependent proposal distribution. For every object, we sample $N_p = \gamma N_s$ particles $\{\mathbf{x}_{p,t}^{(i)}\}_{i=1}^{N_p}$ from the transition prior, where $0 < \gamma < 1$, and the other $N_s - N_p$ particles $\{\mathbf{x}_{q,t}^{(i)}\}_{i=1}^{N_s - N_p}$ are drawn according to

$$\mathbf{r}_{q,t}^{(i)} \sim p(\mathbf{r}_t|\mathbf{z}_t), \quad (4.28)$$

$$\mathbf{v}_{q,t}^{(i)} \sim \mathcal{U}(V_{\min}, V_{\max}), \quad \text{for } k = 1, \quad (4.29)$$

$$\mathbf{v}_{q,t}^{(i)} \sim \mathcal{N}(\mathbf{v}_t | (\mathbf{r}_{q,t}^{(i)} - \hat{\mathbf{r}}_{t-1})T^{-1}, Tq_{1,2}\mathbf{I}), \quad \text{for } k = 2, \quad (4.30)$$

$$\mathbf{s}_{q,t}^{(i)} \sim \mathcal{N}(\mathbf{s}_t | \hat{\mathbf{s}}_{t-1}, q_2T), \quad (4.31)$$

$$I_{q,t}^{(i)} \sim \mathcal{N}(I_t | (1 - \alpha)\hat{I}_{t-1}, q_3T), \quad (4.32)$$

where $\hat{\mathbf{r}}_{t-1}$, $\hat{\mathbf{s}}_{t-1}$, and \hat{I}_{t-1} are MMSE estimates from the object state in the previous frame, and \mathcal{U} denotes the uniform distribution. In this case the region over which the

normalization by H_n in (4.19) is computed is limited to a disk of radius R centered at the centroid position computed using the predicted density (4.25). The radius R is set to the 3-standard-deviation level, where the corresponding variance is estimated from the particle representation of $p(\mathbf{x}_t|k_t, \mathbf{z}_{1:t-1})$ at time t .

The importance function $q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{z}_t)$ in this case is given by

$$q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{z}_t, k_t) = \gamma p(\mathbf{x}_t|\mathbf{x}_{t-1}, k) + (1 - \gamma) \tilde{q}(\mathbf{y}_t|\hat{\mathbf{y}}_{t-1}, \mathbf{z}_t, k) p_s(\mathbf{s}_t|\hat{\mathbf{s}}_{t-1}) p_I(I_t|\hat{I}_{t-1}), \quad (4.33)$$

where

$$\tilde{q}(\mathbf{y}_t|\hat{\mathbf{y}}_{t-1}, \mathbf{z}_t, k = 1) = \tilde{p}(\mathbf{r}_t|\mathbf{z}_t) \mathcal{U}(V_{\min}, V_{\max}), \quad (4.34)$$

$$\tilde{q}(\mathbf{y}_t|\hat{\mathbf{y}}_{t-1}, \mathbf{z}_t, k = 2) = \tilde{p}(\mathbf{r}_t|\mathbf{z}_t) \mathcal{N}(\mathbf{v}_t | (\mathbf{r}_t - \hat{\mathbf{r}}_{t-1}) T^{-1}, T q_{1,2} \mathbf{I}), \quad (4.35)$$

Utilizing the image data, this proposal distribution generates samples from those areas where the likelihood is high, and that are highly consistent with the most recent measurements.

4.4.3 Rao-Blackwellization Approach

As mentioned, in the case of high-dimensional state spaces (in our case $n_x=7$), the SIS becomes inefficient and leads to variance increase of the estimator. However, when the transition and observation models have an analytically tractable structure, the size of the state space can be reduced by analytical marginalization of some of the state variables. This is also called Rao-Blackwellization (RB) [39]. In our applications, for each realization (each MC particle) of the state variable $\mathbf{x}_t = (\mathbf{y}_t, \mathbf{s}_t, I_t)$, we have a linear Gaussian transition and observation model for the intensity, I_t . For such models the optimal solution can be obtained analytically by using the Kalman filter. We therefore combine a (M)PF to compute the distribution of the discrete states $(\mathbf{y}_t, \mathbf{s}_t)$ with a bank of N Kalman filters to compute exactly the distribution of the continuous state I_t . By applying the factorization

$$p(\mathbf{y}_t, \mathbf{s}_t, I_t | \mathbf{z}_{1:t}) = p(I_t | \mathbf{y}_t, \mathbf{s}_t, \mathbf{z}_{1:t}) p(\mathbf{y}_t, \mathbf{s}_t | \mathbf{z}_{1:t}), \quad (4.36)$$

the probability density $p(I_t | \mathbf{y}_t, \mathbf{s}_t, \mathbf{z}_{1:t})$, which is Gaussian, can be computed analytically by applying the Kalman filter:

$$p(I_t | \mathbf{y}_t, \mathbf{s}_t, \mathbf{z}_{1:t}) = \mathcal{N}(I_t | I_{t|t}, P_{t|t}), \quad (4.37)$$

$$I_{t|t-1} = (1 - \alpha) I_{t-1|t-1}, \quad (4.38)$$

$$I_{t|t} = I_{t|t-1} + K_t (Z_t - H_t I_{t|t-1}), \quad (4.39)$$

$$P_{t|t-1} = (1 - \alpha)^2 P_{t-1|t-1} + q_3 T, \quad (4.40)$$

$$P_{t|t} = P_{t|t-1} - K_t H_t P_{t|t-1}, \quad (4.41)$$

$$S_t = H_t P_{t|t-1} H_t^T + R_t, \quad (4.42)$$

$$K_t = P_{t|t-1} H_t^T S_t^{-1}. \quad (4.43)$$

and the vectors H_t and Z_t are formed as

$$H_t = (\dots, a_t(i, j; \mathbf{r}_t, \mathbf{v}_t, \mathbf{s}_t), \dots)^T, \quad (4.44)$$

$$Z_t = (\dots, z(i, j) - b_t(i, j), \dots)^T, \quad (4.45)$$

for all pixels $(i, j) \in C(\mathbf{x}_t)$. Here, $z(i, j)$ denotes the image intensity value at pixel position (i, j) , and $C(\mathbf{x}_t)$ is the region affected by the object with state \mathbf{x}_t , defined as $C(\mathbf{x}_t) = \{(i, j) \in \mathbb{Z}^2 : a_t(i, j; \mathbf{r}_t, \mathbf{v}_t, \mathbf{s}_t) > 0.1\}$. The covariance matrix of the measurement noise, R_t , models the Poisson noise, the main source of noise in optical microscopy imaging, and is given by $R_t = \text{diag}[\dots, h_t(i, j; \mathbf{x}_t), \dots]$. The recursive Bayesian solution is applicable if the statistics of the measurement noise are known.

In summary, we need to estimate only $p(\mathbf{y}_t, \mathbf{s}_t | \mathbf{z}_{1:t})$ using a (M)PF, in a space of reduced dimension, which satisfies the alternative recursion

$$p(\mathbf{y}_t, \mathbf{s}_t | \mathbf{z}_{1:t}) \propto p(\mathbf{y}_{t-1}, \mathbf{s}_{t-1} | \mathbf{z}_{1:t-1}) p(\mathbf{z}_t | \mathbf{y}_t, \mathbf{s}_t, \mathbf{z}_{1:t-1}) p(\mathbf{y}_t, \mathbf{s}_t | \mathbf{y}_{t-1}, \mathbf{s}_{t-1}). \quad (4.46)$$

The likelihood $p(\mathbf{z}_t | \mathbf{y}_t, \mathbf{s}_t, \mathbf{z}_{1:t-1})$ does not simplify to $p(\mathbf{z}_t | \mathbf{y}_t, \mathbf{s}_t)$ because there is a dependency on past values through $I_{0:t}$. For conditionally linear models, we have $Z_t \sim \mathcal{N}(H_t I_{t|t-1}, S_t)$ [39]. Thus, the variable $u_t = (Z_t - H_t I_{t|t-1})^T S_t^{-1} (Z_t - H_t I_{t|t-1})$ is χ_L^2 distributed with L degrees of freedom, where L is the dimension of Z_t . The likelihood $p(\mathbf{z}_t | \mathbf{y}_t, \mathbf{s}_t, \mathbf{z}_{1:t-1})$ in this case is chosen to be $\chi_L^2(u_t)$. In order to compute the threshold π_d , we simulate the background distribution for each MC particle with the vector $Z_{b,t} = (\dots, \zeta(i, j), \dots)^T$ of length L , where $\zeta(i, j) \sim \mathcal{N}(\zeta | 0, b(i, j))$ and introduce a variable $u_{b,t} = (Z_{b,t} - H_t I_{t|t-1})^T U_t^{-1} (Z_{b,t} - H_t I_{t|t-1})$, where $U_t = \text{diag}[\dots, b_t(i, j), \dots]$. Having N_s values of $u_{b,t}$ for each object m , we compute the sample mean $\hat{\pi}(u_{b,t}) = N_s^{-1} \sum_{i=1}^{N_s} \chi_L^2(u_{b,t}^{(i)})$ and similarly the variance $\hat{\sigma}_{\pi,t}^2$. The threshold π_d is defined as $\hat{\pi} + 3\hat{\sigma}_{\pi,t}$ and if $\hat{\pi}(u_t) < \pi_d$ the tracking for the corresponding object is terminated.

The variance of the importance weights for RB(M)PF is lower than for (M)PF. Also, for the same performance, in terms of both accuracy and robustness, fewer MC particles are needed. This is because the dimension of $p(\mathbf{y}_t, \mathbf{s}_t | \mathbf{z}_{1:t})$ is smaller than that of $p(\mathbf{x}_t | \mathbf{z}_{1:t})$. Another reason is that optimal algorithms are used in order to estimate the linear state variables.

4.5 Algorithm Overview

Having described all aspects of the proposed tracking approach in the previous sections, we now give a step-by-step overview of our algorithm, which also summarizes the parameters involved. Apart from parameter setting (steps 1 and 2), which needs to be done by the user depending on the applications, the algorithm is fully automatic.

1. Given image sequences from time-lapse microscopy imaging experiments, specify prior knowledge about object features (Section 4.2.3): σ_{\max} and σ_{\min} (shape), V_{\min} and V_{\max} (velocity range), $q_{1,1}$ and $q_{1,2}$ (motion noise), q_2 (shape noise), q_3 (intensity noise), and matrix Π (motion transition probabilities, Section 4.3). Since the algorithm does not depend critically on these values, rough estimates

suffice, which can be obtained from the image data in advance by limited manual analysis.

2. Specify the free algorithm parameters: N_s (number of MC samples used per object, Sections 4.2.1, 4.2.2), r (signal amplification exponential, Section 4.2.5), h (minimum signal level, Section 4.2.5), and γ (importance sampling mixture ratio, Section 4.4.2). The remaining parameters are precomputed automatically from the image data: α (photobleaching rate, Section 4.2.3), π_d (track termination weight threshold, Section 4.2.5), and N_b (number of samples indicating object birth, Section 4.2.5).

Then, for each successive frame in an image sequence, from the first until the last frame, the algorithm performs the following steps:

3. Perform object detection (Section 4.2.5) in the current frame and initiate a track for each newly detected object, using N_s particles per object and dynamics model, with normally distributed position, velocity, shape, and intensity values, according to the noise levels defined in step 1, and with uniformly distributed weights. Combine the newly detected objects with the existing ones and recompute the mixture weights.
4. Perform the “mixing” of the particles according to (4.21)-(4.23).
5. Propagate all particles according to both dynamics models (Section 4.2.3).
6. Compute the MMSE estimates of the predicted object states using the predicted density (4.25) and draw additional MC samples using the data-dependent importance sampling (Section 4.4.2).
7. Combine the particles resulting from step 5 with the ones from step 6 in proportions defined by parameter γ .
8. Update the particle weights using the marginalization (Section 4.4.1) and Rao-Blackwellization (Section 4.4.3) approaches.
9. Terminate tracks of which the average of the unnormalized particle weights has dropped below the predefined threshold π_d (Section 4.2.5).
10. Update the probability $P(k_t)$ according (4.26) and compute the MMSE estimate for the object positions from the posterior (4.24).

For experimental purposes, the proposed algorithm was implemented in the Java programming language (Sun Microsystems Inc., Santa Clara, CA) as a plugin for ImageJ (National Institutes of Health, Bethesda, MD), a public domain and platform independent image processing program use abundantly in biological image analysis [1]. Running on a regular PC (Intel Core 2 Duo, 2.6 GHz CPU, 4 GB RAM) using the Java Virtual Machine version 1.6, the processing time per object per frame using 10^3 MC particles is about 0.7 sec. In other words, with the current implementation, it takes about 2 hours to track 100 objects over 100 frames. Since tracking can be

done off-line, these numbers are acceptable in practice. For the same accuracy the standard PF [141] requires 2-3 times more MC particles and, because of that, runs 10-20% slower than the proposed RBMPF. We expect that computation times can be reduced by further optimization of the code.

While the algorithm has been described here for tracking in 2D image sequences, it can be easily extended for tracking in 3D and 4D (multichannel 3D) image sequences, by adding the extra dimensions to the state vector and introducing observation models for these additional state elements. For illustrational purposes, we focus in this chapter on applications where, for practical reasons, imaging is done in 2D over time.

4.6 Experimental Results

The proposed algorithm was thoroughly tested using synthetic image data, for which ground truth was available, as well as real biological image data from several time-lapse microscopy studies. In both cases, the dynamics of three different types of intracellular objects were considered, which are representative of the dynamics encountered in practice.

4.6.1 Considered Objects

The first type of objects considered in our experiments are microtubules (MTs). These are filaments composed of α/β -tubulin heterodimers and play the role of conveyer belts in moving chromosomes and vesicles via special attachment proteins. MTs are relatively rigid structures capable of growing and shrinking [37,48]. In this chapter we focus on MT growth events, visualized using fluorescent plus-end-tracking proteins, or +TIPs [137]. Growing MTs mostly exhibit nearly constant linear motion with small accelerations, as described by the second motion model of Section 4.2.3.

As a second type of object we considered vesicles. These are relatively small objects that carry cell products from the Golgi apparatus to the plasma membrane [58]. They are also heavily involved in protein production, as well as helping certain products reach their proper location of operation (such as receptors being incorporated into the plasma membrane). To understand the molecular mechanisms underlying organelle motility and distribution, it is essential to characterize in detail different movement parameters, such as velocities, run lengths, and frequencies of pausing (random-walk type of motion) and switching (from random walk to directed motion). In order to accurately keep track of motion pattern switches, the use of both motion models of Section 4.2.3 is essential.

Finally we considered androgen receptors (ARs). The AR is a ligand-dependent transcription factor that regulates the expression of genes involved in the development and maintenance of the male phenotype and plays a role in the growth of prostate cancer. Quantitative assays such as fluorescence recovery after photobleaching (FRAP) and fluorescence resonance energy transfer (FRET) have been instrumental in the investigation of the behavior of ARs in living cells [46,170]. Similar to many other nuclear factors interacting with DNA, ARs are mobile in the living cell nucleus and dynamically interact with specific binding sites in the promoter/enhancer regions of

the genes they regulate. In addition, ARs are distributed in the nucleus in a typical speckled distribution pattern. It has been hypothesized that these speckles represent ARs bound to promoters [46]. This is supported by confocal microscopy showing a partial overlap between active transcription sites and NR speckles [170]. However, to further investigate the nature of AR speckles, individual object tracking is required. In our experimental data, speckles mostly exhibit restricted Brownian motion, with occasional intervals of more directed motion.

4.6.2 Synthetic Data Experiments

The validation of the proposed tracking framework was first done using synthetic data sets. Three types of motion were modeled according to the dynamics models described in Section 4.2.3. The motion of MTs was modeled using (4.13). The dynamic behavior of vesicles was described by a combination of the models (4.11) and (4.13). Finally, the motion of ARs was modeled using (4.11) only. In the case of vesicles, the switching between the motion patterns was governed by a Markov chain with the following transition matrix, which was learned from the real image data (Section 4.6.3):

$$\Pi = \begin{pmatrix} 0.9 & 0.1 \\ 0.2 & 0.8 \end{pmatrix}. \quad (4.47)$$

Realistic 2D image sequences (Fig. 4.2), consisting of 30–100 frames ($T = 1$ sec.) of 512×512 pixels ($\Delta_x = \Delta_y = 50$ nm) of 20–60 moving objects per frame, were generated for different SNRs in a range around $\text{SNR} = 4$, which previously has been identified as a critical level at which several popular tracking techniques break down [32]. Here, SNR is defined as the difference in intensity between the object (I_o) and background (I_b), divided by the object noise ($\sigma_o = \sqrt{I_o}$) [32].

The parameters of the algorithm were fixed to the following values: $N_s = 10^3$ samples per object, $r = 8$, $h = 20$ (corresponding to $\text{SNR} > 2$), and $\gamma = 0.5$. Elongated MT-like objects were created with the shape parameters set to $\sigma_{\min} = 100$ nm and $\sigma_{\max} = 250$ nm. For the round vesicles and ARs these parameters were set to $\sigma_{\min} = \sigma_{\max} = 100$ nm. Velocities for MTs, and also for vesicles in the directed motion stage, ranged from $V_{\min} = 200$ to $V_{\max} = 700$ nm/sec. The motion, shape, and intensity noise levels were fixed to $q_{1,1} = q_{1,2} = 5000$ nm²/sec², $q_2 = 0.04$, and $q_3 = 100$ nm².

The described motion patterns can be easily distinguished by analyzing the histograms of the displacements and relative angles (Fig. 4.3), where the displacement is defined as the difference in the object position in two successive frames, $\|\Delta\mathbf{r}_t\| = \|\mathbf{r}_{t+1} - \mathbf{r}_t\|$, and the relative angle is the angle between the vectors $\Delta\mathbf{r}_t$ and $\Delta\mathbf{r}_{t+1}$. Instant velocity is calculated as $\|\Delta\mathbf{r}_t\|/T$. The histogram of displacements in the case of ARs follows the χ_2 -distribution and the relative angles are uniformly distributed in the range $[-\pi, \pi]$. The relative angles for the directed, nearly constant velocity motion of MTs are centered around zero, and the displacements are spread over the range of modeled velocities. Finally, the corresponding histograms for the vesicles exhibit a superposition of the two patterns mentioned above.

For all three applications, the estimated parameters of interest (average velocity, run length, drift coefficient, et cetera) are different, but all of them can be derived

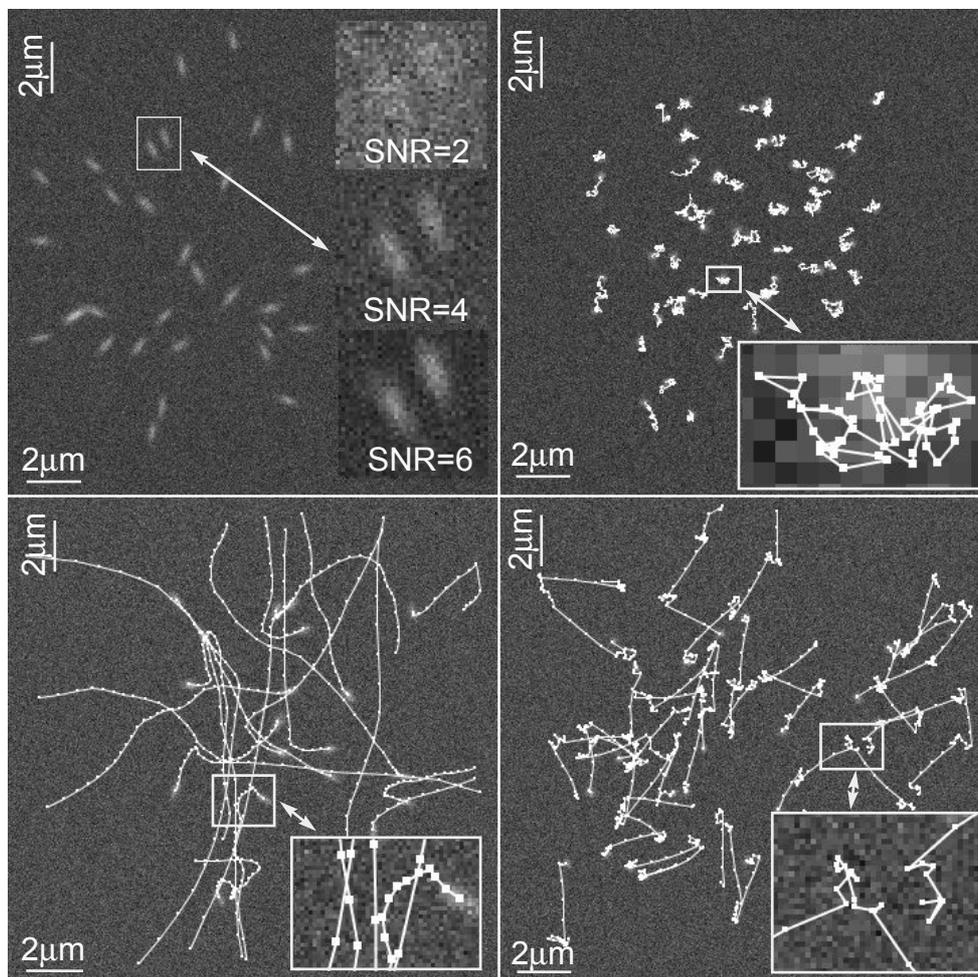


Figure 4.2. Examples of synthetic images used in the experiments giving an impression of object appearance at SNR=4 (top left). The insets show zooms at different SNRs. Also shown are the motion patterns described by the proposed dynamics models for AR-like objects (top right), MTs (bottom left), and vesicles (bottom right). The insets show some of the modeled tracks in more detail.

from the object position in every frame. Hence, accurate position estimation, which is the result of tracking, is the most important criterion for obtaining correct results. Exploiting the available ground truth, we assessed the accuracy of automated tracking using both the standard PF approach and the proposed RBMPF algorithm, by computing the root mean square error (RMSE) in the positions of all correctly tracked objects, as commonly done in trajectory analysis [105, 161]. The rate of correctly tracked objects using PF techniques is close to 100% and is the highest among

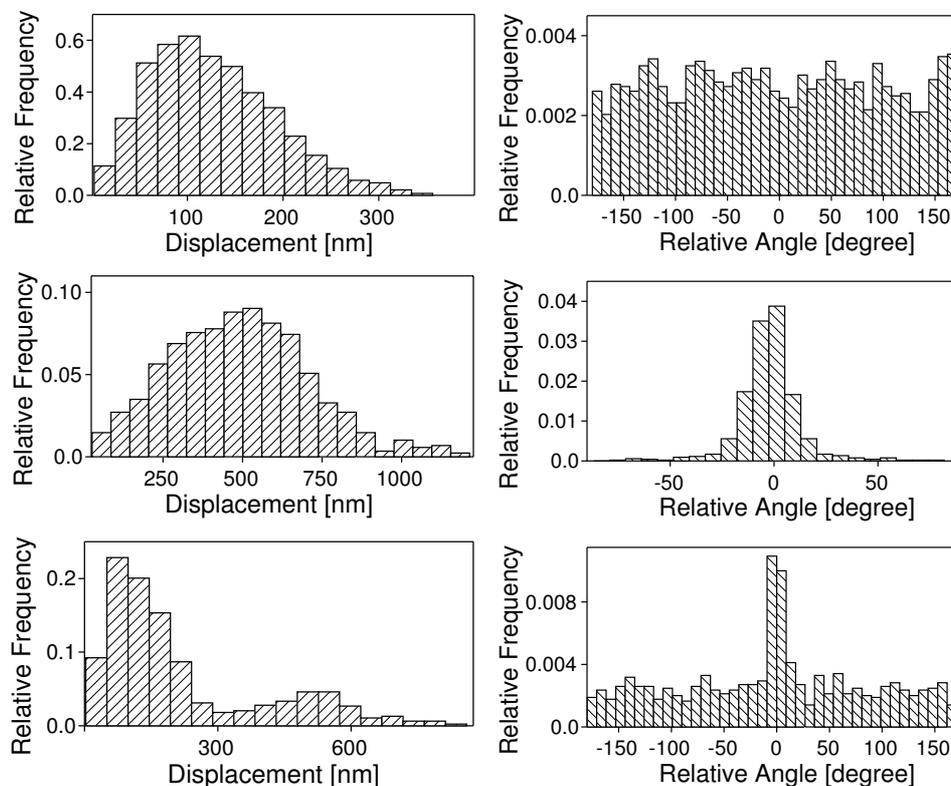


Figure 4.3. Distribution of the displacements and relative angles for the modeled synthetic data according to the described dynamics model for ARs (top row), MTs (middle row), and vesicles (bottom row).

available tracking tools (see [141] for a comparison and sensitivity analysis). The localization errors for both filters are presented in Table 4.1, which also shows the results of manual tracking by five human experts, using the freely available software tool MTrackJ [94]. To keep the workload manageable, manual tracking was limited to the synthetic data sets showing MT-like motion. As can be seen from the table, the errors for automated tracking are approximately 3–4 times lower compared to manual tracking, with our RBMPF algorithm being consistently more accurate (up to 36%) than the standard PF approach.

In the case of vesicles, the results for the standard PF are not shown, as this algorithm uses only a single dynamics model and was unable to follow the switching between the different motion patterns (for most objects the filter lost track). When tracking only a single object, the performance of the standard PF could be theoretically improved, by using an impractically large number of MC samples and substantially increasing the process noise in the dynamics model. However, for multiple object tracking this solution does not work, as the measurement-to-track association would be completely ambiguous. By contrast, the proposed RBMPF algorithm tracked all

Table 4.1. Comparison of the localization errors for manual tracking, the standard PF approach, and the proposed RBMPF algorithm.

SNR	Manual Tracking		ARs		MTs		Vesicles
	RMSE [nm]	StDev [nm]	PF	RBMPF	PF	RBMPF	RBMPF
			RMSE [nm]				
2	130	30	45	43	50	47	40
4	110	20	20	17	39	25	19
6	90	5	19	13	30	20	15

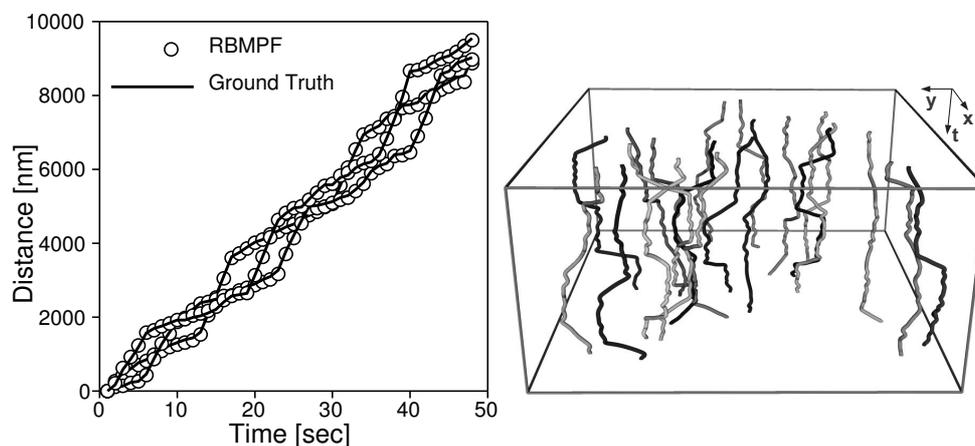


Figure 4.4. Results (three tracks) of automatically tracking vesicles that exhibit switching between fast and slow motion (left). The plot shows traveled distance versus time. Steep slopes represent short stages of fast motion, with almost constant velocity, which are preceded or followed by longer periods of random walk drifts. The tracks are selected from 20 synthetically created tracks (right).

objects without losing track during motion switching (Fig. 4.4).

4.6.3 Real Data Experiments

Real time-lapse fluorescence microscopy image data sets were also obtained for each of the three biological applications considered in this chapter. In all cases, a confocal or widefield microscope was used (Carl Zeiss, Göttingen, Germany) with a Plan Neofluar 40 \times or 100 \times 1.3 NA oil objective. The proposed algorithm was applied to each of the data sets. For these experiments, the object and algorithm parameters were fixed to the same values as in the case of the synthetic data experiments. Since ground truth was not available for the real data, the tracking results were analyzed by expert visual inspection and comparison with manual tracking using MTrackJ [94]. In the following

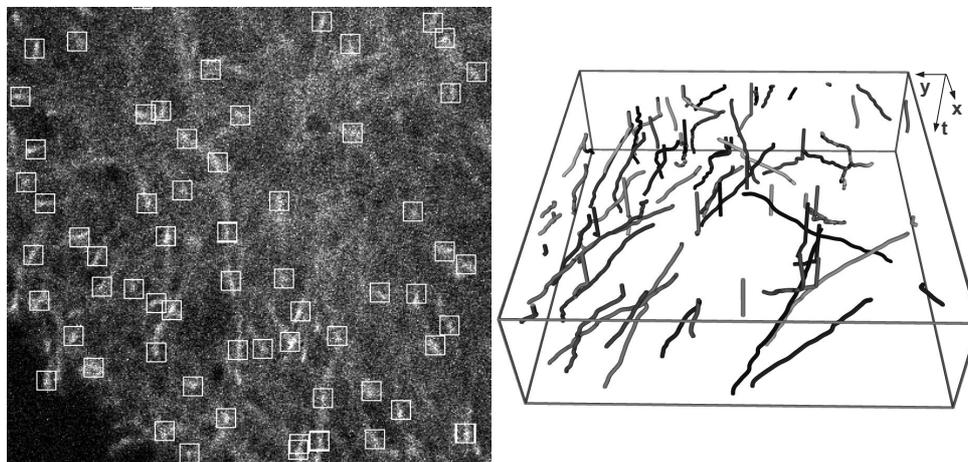


Figure 4.5. Single frame (left) from a time-lapse image data set acquired for studying MT dynamics, with the objects detected by our algorithm marked with white squares, and the results (right) of tracking using the algorithm. All relevant objects were detected and correctly tracked as confirmed by visual inspection.

subsections, we briefly describe the imaging setups and the results obtained for each of the three applications. More details on the cell cultures and imaging conditions can be found in the cited papers.

4.6.3.1 Microtubules

COS-1 cells were cultured and transfected with GFP-tagged proteins as described elsewhere [5,155]. An LSM-510 confocal laser scanning microscope was used to acquire images of GFP+ TIP movements at a rate of 1 frame per 1 or 2 seconds. The image sequences consisted of 30–50 frames of 512×512 pixels of size $75 \times 75 \text{ nm}^2$. Sample tracking results for this application are presented in Fig. 4.5.

The estimation of the accuracy of the proposed algorithm in tracking microtubules was carried out on two typical image sequences by comparison with manual tracking based on 10–20 relevant objects selected by biologists. Distributions of instant velocities and the average velocity per track estimated using our algorithm versus manual tracking are presented in Fig. 4.6. The velocity estimates obtained using the RBMPF algorithm were comparable with those reported previously [155] and did not differ more than 1% from the manual tracking results for the same image data. The main difference between the standard PF approach [141] and the proposed RBMPF algorithm for this application is that the latter is faster. In terms of accuracy, the two algorithms perform comparably.

4.6.3.2 Vesicles

For peroxisome and Rab6 imaging [58], we used HeLa cells and PEX3-GFP fusion, which was a gift from B. Distel [181]. The HeLa cell line is the oldest cell line

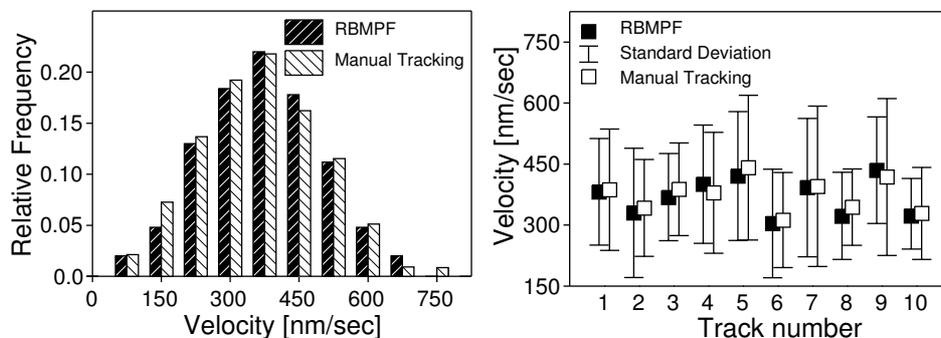


Figure 4.6. Examples of velocity distributions (left) obtained with our RBMPF-based automatic tracking algorithm versus manual tracking applied to real fluorescence microscopy image sequences of growing MTs (data set in Fig. 4.1(a)), and velocity estimation for 10 representative MT objects (right). Shown are the mean values (black or white squares) and ± 1 standard deviation (bars) of the estimates.

and is widely used for many different studies. Many variants of the HeLa cell line exist, among which HeLa-R, with a so-called “round” phenotype, and HeLa-L, with a “long” phenotype. They differ from each other by several features, such as LL5 cortical protein distribution [81], and have a strikingly different cell shape (round and elongated, respectively). We used HeLa-L cells to study the kinetic properties of the Rab6-vesicles movements (work in progress) and HeLa-R cells to study microtubule dynamics [98], microtubules and cell cortex crosstalk [81], and exocytosis [58]. Time-lapse images were acquired on an Axiovert 200M inverted microscope at a rate of 0.83 frames per second. The image sequences consisted of 100 frames of 1344×1024 pixels of size 64×64 nm². Sample tracking results for this application are presented in Fig. 4.7.

Validation of the RBMPF algorithm for vesicle tracking in HeLa-R and HeLa-L cells was done by comparison with manual tracking for 30 vesicles that were visible long enough (at least 40 frames) to clearly exhibit typical switching patterns. Distributions of the displacements and relative angles of moving vesicles in both types of HeLa cells are shown in Fig. 4.8. In order to estimate vesicle velocity during the fast motion stage, a threshold of $0.3 \mu\text{m}/\text{sec}$ was introduced to separate motion stages, as in previous studies [58]. The velocity estimates, as well as the average ratio of the time that a vesicle is in the fast motion stage to the total track time, are given in Table 4.2, for both automatic and manual tracking. The table also shows the probabilities of switching between the two motion stages, which is another important parameter used in modeling and studying intracellular dynamics, and in our model corresponds to the values of p_{12} and p_{21} of the transition matrix Π . In contrast with our previous algorithm [141], the RBMPF algorithm was capable of catching the transitions between the two types of motion (see Fig. 4.9), and the computed parameters were in good agreement with manually obtained values.

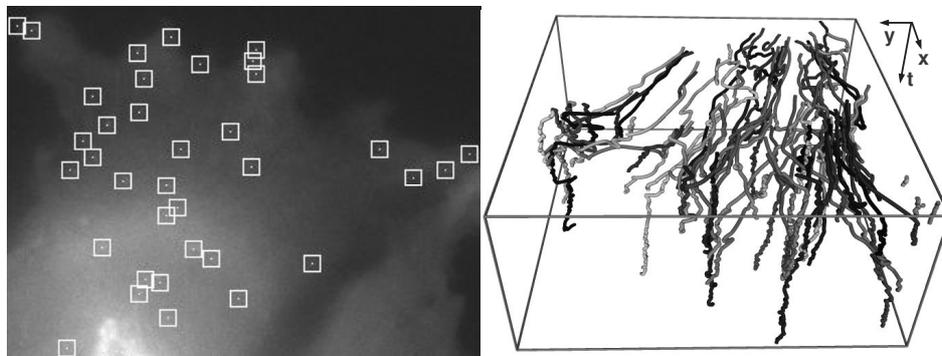


Figure 4.7. Single frame (left) from a time-lapse image data set acquired for studying vesicle dynamics, with the objects detected by our algorithm marked with white squares, and the results (right) of tracking using the algorithm. All relevant objects were detected and correctly tracked as confirmed by visual inspection.

Table 4.2. Comparison of estimates of velocity [$\mu\text{m}/\text{sec}$], the average ratio of fast motion time to total track time, and the probability of switching between the two motion stages of vesicles in HeLa-R and HeLa-L cells, based on manual tracking versus automatic tracking using the proposed RBMPF algorithm.

Estimate	Manual Tracking		Automatic Tracking	
	HeLa-R	HeLa-L	HeLa-R	HeLa-L
Velocity	0.72 ± 0.23	0.83 ± 0.35	0.75 ± 0.28	0.80 ± 0.31
Fast motion ratio	0.23	0.27	0.21	0.28
Slow to fast probability	0.09	0.10	0.08	0.12
Fast to slow probability	0.27	0.22	0.28	0.22

4.6.3.3 Androgen Receptors

For AR imaging [170], YFP emission was detected in Hep3B cells expressing YFP and CFP (yellow and cyan variants of GFP respectively) double-tagged AR. An LSM-510 microscope was used to acquire images at a rate of 1 frame per 12.6 seconds. The resulting image sequences consisted of 80–180 frames of 512×512 pixels of size $75 \times 75 \text{ nm}^2$. Sample tracking results for this application are presented in Fig. 4.10.

Because of the previous lack of robust tracking tools and the considerable labor involved in manual analysis of this type of image data, actual analysis of the data had not been performed prior to the present study. Quantitative performance analysis of the proposed algorithm was done in comparison with expert manual tracking of a limited number of 16 relevant objects. The displacement and relative angle his-

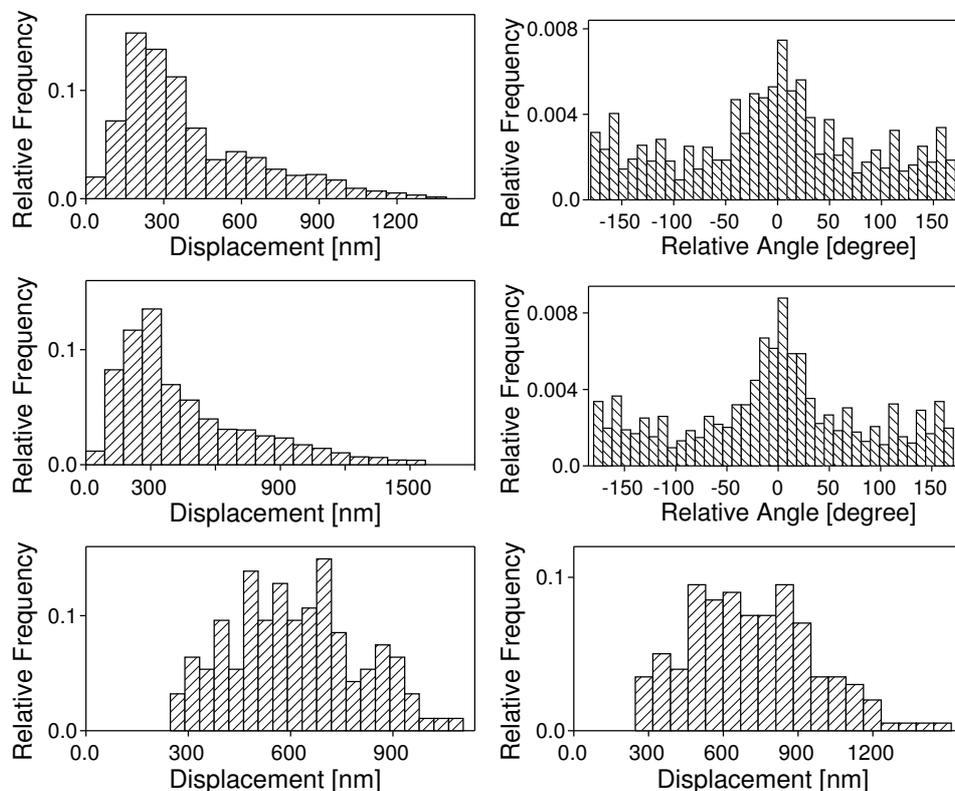


Figure 4.8. Distribution of the displacements and relative angles for 30 automatically obtained vesicle tracks for HeLa-R (top row) and HeLa-L (middle row) cells, and the filtered (thresholded) displacement histograms (bottom row), used for velocity analysis during fast motion stages for HeLa-R (left) and HeLa-L (right) cells.

tograms obtained by automatic and manual means are presented in Fig. 4.11 and show no statistically significant difference. Estimation of the average displacement per frame was done by computing the sample mean and variance. The estimates are $123 \pm 67\text{nm}$ and $128 \pm 81\text{nm}$ for automatic and manual tracking, respectively. The proposed automatic tracker correctly reproduced the relative angle distribution, and, considering the shape of that distribution (Fig. 4.11), revealed the fact that the objects under consideration do not freely move but rather seem to be attached to immobile structures. This attachment constrains the behavior and reduces the number of degrees of freedom: every time the object moves, its displacement in the next frame will be rather in the opposite direction. The displacement and relative angle time series for 3 tracks are shown in Fig. 4.12. Other parameters, such as the size, shape, and intensity of single speckles, as well as their behavior over time can also be analyzed automatically by our algorithm and will be the subject of future studies.

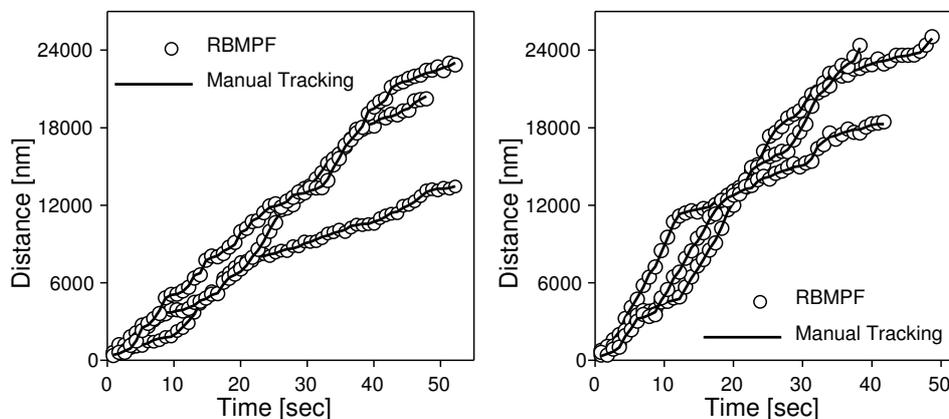


Figure 4.9. Results (three tracks) of automatically tracking vesicles that exhibit switching between fast and slow motion for HeLa-R (left) and HeLa-L (right) cells. The plots show traveled distance versus time. Steep slopes represent short stages of fast motion, with almost constant velocity, which are preceded or followed by longer periods of random walk drifts.

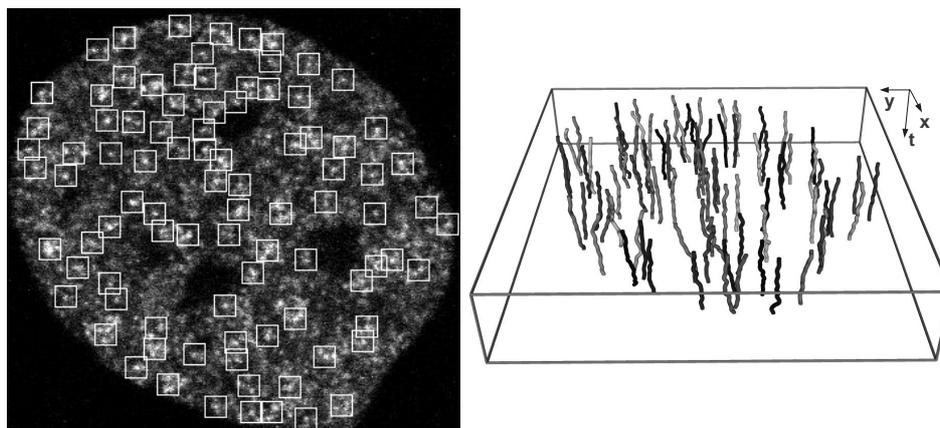


Figure 4.10. Single frame (left) from a time-lapse image data set acquired for studying AR dynamics, with the objects detected by our algorithm marked with white squares, and the results (right) of tracking using the algorithm. All objects with sufficiently high local SNR as determined by the h -parameter (Section 4.2.5) were detected and correctly tracked as confirmed by visual inspection.

4.7 Discussion and Conclusions

In this chapter we have proposed a novel algorithm for simultaneous tracking of many nanoscale objects in time-lapse fluorescence microscopy image data sets. The algorithm, which is built within a Bayesian tracking framework, shows several important improvements compared to our previous work [141, 145]. Tracking accuracy is im-

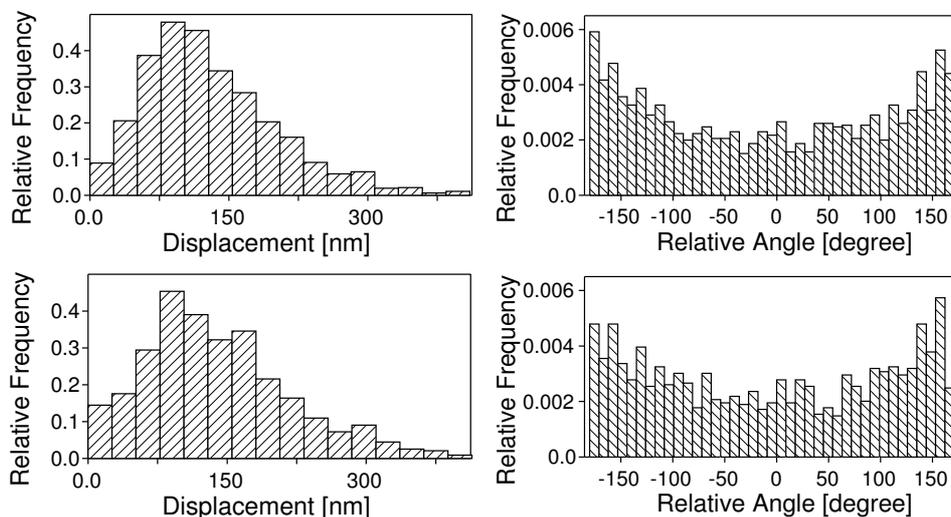


Figure 4.11. Displacements and relative angle distributions in the case of automatic (top) and manual (bottom) analysis of AR dynamics.

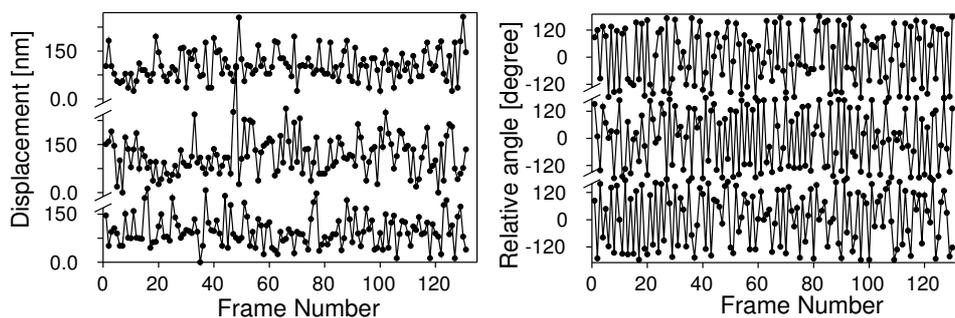


Figure 4.12. Examples of the velocity (left) and relative angle (right) time series for three automatically analyzed AR tracks.

proved by using marginalization of the filtering distribution and one of the state variables, for which the optimal solution (the Kalman filter) is used. In addition, improved robustness is achieved by integrating a jump Markov system into the framework, which allows the use of multiple dynamics models for object motion prediction. Since common Bayesian tracking algorithms are designed to deal with only one specific type of motion, they often fail when used for real biological applications, where usually more complex motion patterns need to be analyzed.

The proposed algorithm was tested on synthetic image data as well as on real time-lapse fluorescence microscopy data acquired for studying the dynamics of three different types of intracellular objects: microtubules, vesicles, and androgen receptors. Results from the synthetic data experiments clearly showed the superiority of the proposed algorithm over manual tracking as well as our previous Bayesian

tracking approaches, which were already demonstrated to be superior to alternative non-Bayesian tracking algorithms. The real data experiments confirmed the validity of the tracking results produced by the proposed algorithm. Based on these results we have started using our algorithm for attacking specific biological questions.