

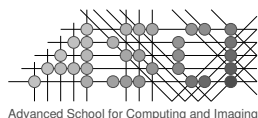
# Particle Filtering Methods for Subcellular Motion Analysis

Ihor Smal

## Colophon

This book was typeset by the author using  $\text{\LaTeX}2_{\epsilon}$ . The main body of the text was set using a 10-points Computer Modern Roman font. All graphics and images were included formatted as Encapsulated PostScript ( $^{\text{TM}}$  Adobe Systems Incorporated). The final PostScript output was converted to Portable Document Format (PDF) and transferred to film for printing.

Cover design by the author using the Persistence of Vision Raytracer (POV-Ray 3.6), a high-quality, totally free tool for creating three-dimensional graphics. The graphics on the cover symbolically represents some of the constituents of the Bayesian estimation framework: measurements (fluorescence microscopy images), a posterior probability density function approximated by a set of “particles”, estimated tracks and randomness (symbolized by dices).



Advanced School for Computing and Imaging

This work was carried out in the ASCI graduate school.  
ASCI dissertation series number 177.

The research described in this thesis was carried out at the Erasmus MC – University Medical Center Rotterdam (Rotterdam, the Netherlands). This work was financially supported by the Netherlands Organization for Scientific Research (NWO) through VIDI-grant 639.022.401.

Financial support for the publication of this thesis was kindly provided by the Department of Radiology of Erasmus MC – University Medical Center Rotterdam, and the Erasmus University Rotterdam, the Netherlands.

Copyright © 2009 by Ihor Smal. All rights reserved. No part of this publication may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopy, recording, or any information storage and retrieval system, without permission in writing from the author.

ISBN 978-90-9024313-9

Printed by Ipskamp Drukkers

# Particle Filtering Methods for Subcellular Motion Analysis

Probabilistische methoden voor subcellulaire bewegingsanalyse

## Proefschrift

ter verkrijging van de graad van doctor aan de  
Erasmus Universiteit Rotterdam  
op gezag van de  
rector magnificus

Prof.dr. S.W.J. Lamberts

en volgens besluit van het College voor Promoties.

De openbare verdediging zal plaatsvinden op  
woensdag 10 juni 2009 om 9:45 uur

door

**Ihor Smal**

geboren te Lviv, Oekraïne



## **Promotiecommissie**

Promotor: **Prof.dr. W.J. Niessen**

Overige leden: **Prof.dr.ir. H.H. Weinans**  
**Dr.ir. N.J. Galjart**  
**Dr. J.-C. Olivo-Marin**

Copromotor: **Dr.ir. H.W. Meijering**

---

## Preface

---

*Do not be desirous of having things done quickly. Do not look at small advantages. Desire to have things done quickly prevents their being done thoroughly. Looking at small advantages prevents great affairs from being accomplished.*

— CONFUCIUS (551–479 B.C.)

This thesis describes the research I carried out as part of my Ph.D. study at the Erasmus University Rotterdam. The actual research was done at the Erasmus MC – University Medical Center Rotterdam, within the Biomedical Imaging Group Rotterdam (BIGR). Without the support of the BIGR members and the people from the Departments of Medical Informatics and Radiology, my staying and research would not be as pleasant and enjoyable as they were.

I would like to acknowledge the people who have made the completion of this dissertation possible. First of all I extend my heartfelt gratitude to my promotor Prof. Wiro Niessen and co-promotor Erik Meijering, for offering me a Ph.D. position in the relatively “young” at that time but full of potential BIGR. Your vital encouragement and support, understanding and assistance made my scientific research go very smooth and without any frustrations. Even the criticism, which was always constructive, encouraged and challenged me through these years. Both of you guided me through the dissertation process, never accepting less than my best efforts. The scientific freedom that you gave me in the very beginning allowed me to study different approaches and search for new possibilities that could be used to achieve our goals – find solutions to tracking problems in bioimaging. Rather than stick with fine-tuning of the existing, more or less working methods for our application, that freedom led me to acquaintance with Particle Filters, which are the primary subject of this thesis. At that time those techniques were recently introduced in some other fields, but had not been applied for biological applications. The design and adaptation of such PF-based techniques to subcellular motion analysis kept me busy for the last four years and all the findings during that time are presented in this book.

In view of the biological datasets used in this thesis, several acknowledgments are in order. I would like to thank people from the Department of Cell Biology and the Department of Pathology, namely Niels Galjart, Katharina Draegestein, Anna Akhmanova, Ilya Grigoriev, Adriaan Houtsmuller and Martin van Royen. It was a great experience to work with you and transform that collaboration into a few journal

papers, some of which are published, others are underway. During the development of the methods described in this book, it was extremely interesting and useful to discuss lots of practical aspects of fluorescence microscopy imaging and the background of all the biological experiments with experts like you.

I would like to acknowledge Marco Loog and Hans Driessen, who are not directly related to the Erasmus MC, but who had some influence on the work presented in this book. The brainstorm sessions and discussions with you were always encouraging and interesting. It is a pity that only a few of those ideas have materialized, mainly due to limited time, hopefully there will be more time for that in the future.

I am indebted to all the people from BGR: Albert, Azadeh, Coert, Danijela, Empar, Esben, Fedde, Henri, Hortense, Jifke, Leijla, Marcel, Marcel, Marius, Marleen, Mart, Michiel, Nóra, Oleh, Rashindra, Reinhard, Renske, Rik, Sennay, Stefan, Theo. Staying in such scientifically fertile and socially enjoyable environment was of great help during my Ph.D. trajectory. Thanks also go to all the staff of the Departments of Medical Informatics and Radiology, especially to Désirée de Jong and Petra 't Hart-Assems, who made my work-related life easier by taking care of all the bureaucratic issues.

Most especially, I wish to thank my parents, Olha and Vasyl and my brother Roman, for their support through all these years, for their encouragement and patient love that enabled me to achieve this goal. Finally, very special thanks go to my wife Nataliya. I feel very lucky to have all of you in my life, therefore I wish to thank God, who made all these things possible.

Ihor Smal  
Rotterdam, February 2009

---

---

# Contents

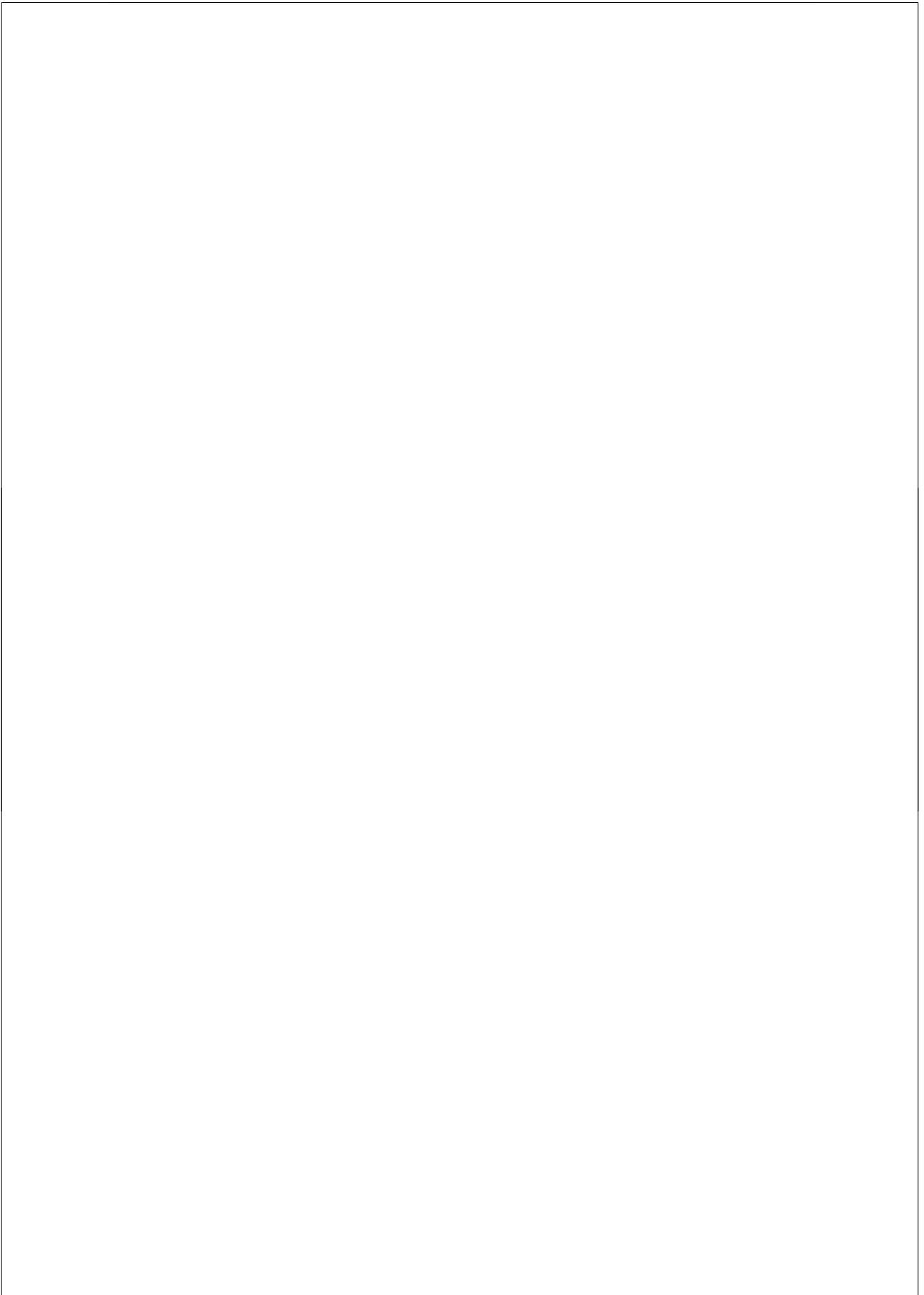
---

<b>Colophon</b>	<b>ii</b>
<b>Preface</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Studying Intracellular Dynamics . . . . .	1
1.2 Fundamental Limitations in Microscopy . . . . .	3
1.3 Tracking in Fluorescence Microscopy . . . . .	6
1.3.1 Image Preprocessing . . . . .	6
1.3.2 Object Localization . . . . .	6
1.3.3 Solving the Correspondence Problem . . . . .	8
1.3.4 Probabilistic Methods for Tracking . . . . .	11
1.4 Analyzing Tracking Results . . . . .	13
1.5 Thesis Outline . . . . .	15
<b>2 Quantitative Comparison of Spot Detection Methods in Fluorescence Microscopy</b>	<b>17</b>
2.1 Introduction . . . . .	18
2.2 Detection Framework for Fluorescence Microscopy . . . . .	19
2.2.1 Image formation . . . . .	19
2.2.2 Detection Framework . . . . .	22
2.3 Detection Methods . . . . .	23
2.3.1 Noise Reduction . . . . .	23
2.3.2 Unsupervised Signal Enhancement . . . . .	25
2.3.3 Supervised Signal Enhancement . . . . .	29
2.3.4 Signal Thresholding and Performance Measures . . . . .	32
2.4 Experimental results . . . . .	33
2.4.1 Evaluation on Synthetic Image Data . . . . .	33
2.4.2 Evaluation on Real Image Data . . . . .	46
2.5 Discussion and Conclusions . . . . .	50
<b>3 Particle Filtering for Multiple Object Tracking in Dynamic Fluorescence Microscopy Images: Application to Microtubule Growth Analysis</b>	<b>53</b>
3.1 Introduction . . . . .	54
3.2 Microtubule Growth Analysis . . . . .	57

3.3	Tracking Framework . . . . .	58
3.3.1	Nonlinear Bayesian Tracking . . . . .	58
3.3.2	Particle Filtering Methods . . . . .	59
3.3.3	Multi-Modality and Mixture Tracking . . . . .	60
3.4	Tailoring the Framework . . . . .	61
3.4.1	State-Space and Dynamic Model . . . . .	61
3.4.2	Object Interactions and Markov Random Field . . . . .	63
3.4.3	Observation Model and Likelihood . . . . .	63
3.4.4	Hierarchical Searching . . . . .	66
3.4.5	Measurement Gating . . . . .	67
3.4.6	Data-Dependent Sampling . . . . .	68
3.4.7	Clustering and Track Management . . . . .	69
3.4.8	Initialization and Track Initiation . . . . .	70
3.5	Experimental Results . . . . .	71
3.5.1	Evaluation on Synthetic Data . . . . .	71
3.5.2	Evaluation on Real Data . . . . .	77
3.6	Discussion and Conclusions . . . . .	80
<b>4</b>	<b>Multiple Object Tracking in Molecular Bioimaging by Rao-Blackwellized Marginal Particle Filtering</b>	<b>85</b>
4.1	Introduction . . . . .	86
4.2	Probabilistic Tracking Framework . . . . .	88
4.2.1	Particle Filtering Approach . . . . .	88
4.2.2	Multiple Object Tracking . . . . .	89
4.2.3	Dynamics Models . . . . .	90
4.2.4	Observation Model . . . . .	92
4.2.5	Track Management . . . . .	93
4.3	Incorporating Multiple Dynamics . . . . .	94
4.4	Applying Marginalization Concepts . . . . .	95
4.4.1	Filtering Distribution Marginalization . . . . .	96
4.4.2	Data-Dependent Sampling . . . . .	96
4.4.3	Rao-Blackwellization Approach . . . . .	97
4.5	Algorithm Overview . . . . .	98
4.6	Experimental Results . . . . .	100
4.6.1	Considered Objects . . . . .	100
4.6.2	Synthetic Data Experiments . . . . .	101
4.6.3	Real Data Experiments . . . . .	104
4.7	Discussion and Conclusions . . . . .	109
<b>5</b>	<b>Microtubule Dynamics Analysis using Kymographs and Variable-Rate Particle Filters</b>	<b>113</b>
5.1	Introduction . . . . .	114
5.2	Methods . . . . .	115
5.2.1	<i>In Vitro</i> Microtubule Dynamics Model . . . . .	115
5.2.2	Imaging Technique and Kymographs . . . . .	116
5.2.3	Edge Preserving Smoothing . . . . .	118



5.2.4	Variable-Rate Particle Filtering . . . . .	120
5.2.5	Multiscale Trend Analysis . . . . .	125
5.3	Experimental Results . . . . .	126
5.3.1	Evaluation on Synthetic Data . . . . .	126
5.3.2	Evaluation on Real Data . . . . .	131
5.4	Discussion and Conclusions . . . . .	131
<b>6</b>	<b>Summary</b>	<b>133</b>
	<b>Bibliography</b>	<b>137</b>
	<b>Samenvatting</b>	<b>147</b>
	<b>PhD Portfolio</b>	<b>151</b>
	<b>Publications</b>	<b>153</b>
	<b>Curriculum Vitae</b>	<b>155</b>



---

# Introduction

---

*There are two possible outcomes: If the result confirms the hypothesis, then you've made a measurement. If the result is contrary to the hypothesis, then you've made a discovery.*

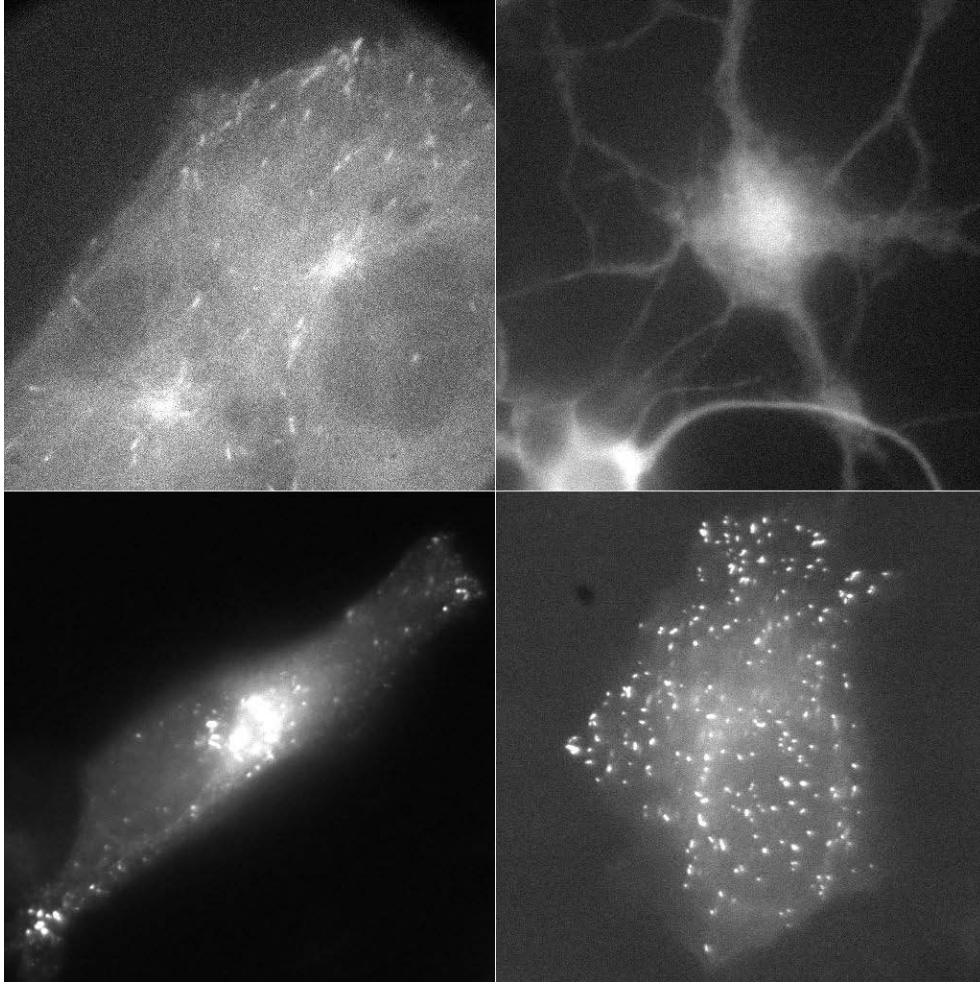
— ENRICO FERMI (1901–1954)

## 1.1 Studying Intracellular Dynamics

The past decades have witnessed development of groundbreaking tools and techniques for imaging and studying cellular and intracellular structures and processes. The advent of confocal microscopy in the early sixties accompanied by discovery of fluorescent proteins has triggered the development of new imaging techniques and revolutionized the way biologists study cells and the way they function. Currently, fluorescence microscopy imaging is still the most important and frequently used tool for studying intracellular dynamics with a high spatial and temporal resolution. Proper understanding of cellular and molecular processes is of great interest to academic researches as well as pharmaceutical industries. The possibility to influence those processes in a controlled way is a prerequisite to combat diseases and improve human health care, which will have profound social and economic impact.

In fluorescence microscopy, the studying of the dynamical processes within a cell is usually done by labeling intracellular structures of interest with fluorescent proteins and following them in time using time-lapse imaging (see Fig. 1.1). The observed dynamical processes can be either studied qualitatively or using some quantitative measures that characterize intracellular behavior. Tracking of subcellular structures in time leads to creating of so called life histories, from which motion parameters such as velocity, acceleration and/or intensity changes in time can be easily estimated.

In practice, fluorescence microscopy, which in many laboratories become a universal tool for studying cellular and intracellular life, has some inherent limitations. One of them is autofluorescence. Autofluorescence describes the emission of fluorescence from naturally fluorescent molecules other than the fluorophore of interest. In fluorescence microscopy imaging, it is a significant source of background noise in images, which can be reduced either by special sample preparation or by background subtraction using image processing methods [185]. Another limiting factor is photobleaching. Photobleaching is the photochemical destruction of a fluorophore, which complicates the observation of fluorescent molecules, since they will eventually be destroyed by the



**Figure 1.1.** Examples of images acquired for different biological studies based on GFP labeling and fluorescence microscopy. The images are single frames from 2D time-lapse studies of activity of microtubule plus-ends (top left), microtubule plus-ends in neurons (top right), Rab5 (bottom left) and peroxisomes (bottom right).

light exposure necessary to stimulate them into fluorescing. This is especially problematic in time-lapse microscopy, where the fluorescence signal is imaged in time, but due to the photobleaching fades permanently lowering the image quality. On the positive side, however, these limiting phenomena serve as a basis for many advanced fluorescence measurement techniques. An example of this is fluorescence recovery after photobleaching (FRAP), which allows to determine diffusion coefficients, binding and dissociation rates [156, 185]. With this technique, the region of interest within a cell is photobleached, and the subsequent recovery in the bleached region as a result of movement of nonbleached fluorescent molecules from the surrounding areas is ob-

served and studied. By measuring the extent and speed to which this recovery occurs, conclusions can be drawn about diffusion of proteins within a membrane or protein turnover in complexes.

To study the localization of the photobleached molecules, sometimes a second fluorophore that remains visible during the imaging is added to the target subcellular structure. This process is called fluorescence localization after photobleaching (FLAP). Another technique, complementary to FRAP, is termed fluorescence loss in photobleaching (FLIP) [156,185]. This procedure involves repeated photobleaching of a cell region, which leads to permanent loss of the fluorescence light signal throughout the whole cell. If the loss is indeed observed, it indicates that free exchange between the molecules occurred between the bleached region and the rest of the cell. Otherwise, if there is no loss in the signal over the whole cell, the molecules in the bleached region are isolated and specifically localized in distinct cellular compartments.

A relatively new technique, which is used to study protein interaction, is fluorescence energy transfer (FRET) [156,185]. FRET involves the radiationless transfer of energy from a donor fluorophore to an appropriately positioned acceptor fluorophore in a nanometer range. Such colocalization techniques are used to reveal functionally related molecules, and map the potential protein-to-protein interactions with high precision providing better understanding of how the intracellular dynamics is regulated, and thereby establishing its relationship to important disease processes. Other frequently used techniques are fluorescence lifetime imaging (FLIM), fluorescence in situ hybridization (FISH) and fluorescence ratio imaging (RI) [185].

Current biological studies using time-lapse fluorescence microscopy imaging require analysis of huge amounts of image data. A large-scale analysis of the dynamics of subcellular objects such as microtubules or vesicles cannot possibly be done without automatic tracking tools. The possibilities to study new aspects of the intracellular dynamics opened by modern imaging tools in combination with advanced image processing techniques impose high standards on robustness and accuracy of the tracking techniques for quantitative motion analysis. Moreover, there is demand for computationally fast methods that are capable of processing large amounts of data, which are typical for high-throughput experiments.

Tracking of multiple objects in biological image data is a challenging problem largely due to poor imaging conditions and complicated motion scenarios. Existing tracking algorithms for this purpose often do not provide sufficient robustness and/or are computationally expensive. By using such automatic tracking tools, biologists also eliminate the bias and possibly the systematic errors they introduce during manual tracking due to intuitive selection of relatively small subsets of objects of interest that are either nicely imaged or exhibit typical or expected motion patterns. Thus, automatic tracking methods capable of following large number of objects in time and classifying their dynamics, are of major interest.

## 1.2 Fundamental Limitations in Microscopy

In light microscopy, several factors complicate quantitative data analysis. In practice, careful design of experiments, the imaging system, and selection of appropriate tools

for the analysis can greatly reduce the influence of some of them. Nonetheless, light microscopy also has fundamental limitations that cannot be overcome and, most of the time, in real experiments biologists are inevitably facing those barriers. One of them is the limited spatial resolution of the microscope – there is a fundamental maximum to the resolution of any optical system due to diffraction. The diffraction limit depends on the emission wavelength, the numerical aperture of the objective lens, and defines the microscope point-spread function (PSF), which describes the response of an imaging system to a point light source. The Fraunhofer-diffraction limited PSF (normalized to unit magnitude at the origin) of a wide-field fluorescence microscope (WFFM) with circular aperture is given by [59]

$$\text{PSF}(r, z) = \left| \int_0^1 2J_0(\alpha r \rho) \exp(-2i\gamma z \rho^2) \rho d\rho \right|^2,$$

where

$$\alpha = \frac{2\pi \text{NA}}{\lambda} \quad \text{and} \quad \gamma = \frac{\pi \text{NA}^2}{2\lambda n},$$

and  $r = \sqrt{x^2 + y^2}$  denotes the radial distance to the optical axis,  $z$  is the axial distance to the focal plane,  $i$  the imaginary unit number,  $J_0$  the zero-order Bessel function of the first kind, NA the numerical aperture of the objective lens,  $n$  the refractive index of the sample medium and  $\lambda$  the wavelength of the light emitted by the specimen. For a laser scanning confocal microscope (LSCM), the PSF is a combination of the excitation and emission intensity distributions. In the case of ideal confocality (infinitely small pinhole size) and assuming that the wavelengths of the emission and excitation light are approximately the same, the LSCM PSF reduces to the product of two WFFM PSFs [190]. In practice, a Gaussian approximation of the PSF is used, which is favored for computational reasons but is nevertheless almost as accurate as more complicated PSF models [55, 190]. The approximation (normalized to unit magnitude at the origin) is given by

$$\text{PSF}_g(r, z) = \exp\left(-\frac{r^2}{2\sigma_r^2} - \frac{z^2}{2\sigma_z^2}\right),$$

where  $\sigma_r^2$  and  $\sigma_z^2$  (for a confocal microscope) are given by [190]

$$\sigma_r = 0.16 \frac{\lambda}{\text{NA}} \quad \text{and} \quad \sigma_z = 0.55 \frac{\lambda n}{\text{NA}^2}.$$

In this case, for noise-free images, the lateral and axial distances of resolution,  $d_{xy}^R$  and  $d_z^R$ , for equally bright fluorescent tags is given by the Rayleigh distances [69]

$$d_{xy}^R = 0.56 \frac{\lambda}{\text{NA}} \quad \text{and} \quad d_z^R = 1.5 \frac{\lambda n}{\text{NA}^2}.$$

For typical microscope setups the lateral resolution is on the order of 200 nm, and the axial resolution, which is always worse, is on the order of 600 nm. This resolution barrier is always encountered in experiments where subcellular structures

(microtubules, vesicles, etc.) are studied. Due to subresolution sizes of those objects ( $< 10 - 20$  nm), they appear in the images as blurred spots. Attempts to overcome these limits range from engineering of new optical systems, such as multiphoton microscopy, stimulated emission and depletion (STED) microscopy, or 4Pi microscopy, to applying sophisticated post-acquisition computational analysis methods that do not require any modifications of the imaging system [60]. In the latter case, deconvolution algorithms and super-resolution methods are used, which necessarily exploit the prior knowledge about the optical system and/or the image formation process. A number of advanced deconvolution methods are available for image restoration in microscopy imaging [24, 65, 109, 129]. While there are suggestions in the literature to always deconvolve the image data if possible [24], the question whether deconvolution is beneficial in fact depends on the application. Most reports on tracking of subcellular structures do not mention the use of deconvolution, because the localization of such diffraction limited objects can be done with much higher accuracy and precision than the resolution of the imaging system using super-resolution methods [2, 35, 118, 162]. On the other hand, these two (deconvolution and super-resolution) approaches for post-acquisition image enhancement are not completely independent. Some super-resolution algorithms, for instance, are based on fitting (a model) of the PSF – to some degree this is in fact deconvolution, carried out implicitly in the process.

The second factor that complicates the data analysis is noise, which is a stochastic phenomenon that cannot be compensated for, contrary to systematic distortions such as blurring. In light microscopy, the imaging is commonly done using a charge-coupled device (CCD) camera, which is a semiconductor device that converts the incoming light photons first to electrical charges and then to voltages which are read out from the device, quantized and stored as a digital image. Unfortunately, every step of this imaging process is influenced by different noise sources: photon noise, thermal noise (dark current and hot pixels), readout noise (on-chip electronic noise) and quantization noise [179]. Photon noise, which is due to the quantum nature of light, follows the Poisson distribution and is signal/amplitude dependent. Since the Poisson distribution approaches the Gaussian distribution for large numbers, the photon noise in a signal will approach the normal distribution for large numbers of collected photons. Thermal noise is also Poisson distributed but can be greatly reduced by cooling the CCD chip. Readout noise, the influence of which becomes significant only for high readout rates ( $> 1$  MHz), is caused by the on-chip electronics. This source of noise is Gaussian distributed and independent of the signal. Quantization noise is caused by the conversion of the analog signal (voltage) to digital representation. This noise is additive, uniformly distributed, and with modern analog-to-digital converters is very low and usually ignored. In practice, the influence of all of these noise sources (except for the photon noise) can be made negligible by proper electronic design and careful operation conditions. Thus, photon noise is the main and fundamental limiting factor that defines the signal-to-noise ratio (SNR) of the image data in microscopy imaging. Low SNRs are especially typical in live-cell fluorescence microscopy, since in most experiments the imaged light signal is quite weak – high excitation light rapidly quenches fluorescence and may disturb intracellular processes being studied.

A final complicating factor worth mentioning here is the large variability of biological image data. This especially complicates the development of universal automatic



methods for quantitative image analysis. In molecular biology, which is a highly experimental field, the absence of standardization in the acquisition and data storage protocols leads to image data of strongly varying quality, even within one type of experiments. This heterogeneity negatively influences the validation of the automatic techniques for studying the intracellular processes and lowers the reproducibility of new results and findings. All these factors put high demand on the design of automated image analysis techniques. This is in contrast with medical investigations, where routine clinical studies are based on standardized imaging protocols, leading to more consistent image quality.

### 1.3 Tracking in Fluorescence Microscopy

The quantitative analysis of time-lapse image sequences that visualize intracellular processes usually requires tracking of multiple objects over time. The majority of the automated tracking techniques described in the literature and available in practice process the data by following a few well established subsequent steps: preprocessing the image data, detecting the objects of interest independently in every image frame, and creating the trajectories by linking the sets of detected objects in subsequent frames. Extracted trajectories are further used for estimation of important parameters that characterize the intracellular dynamics.

#### 1.3.1 Image Preprocessing

The main purpose of preprocessing is to enhance the image quality and, if necessary, compensate for global cell motion. Recent comparative studies demonstrated that the accuracy of commonly used tracking methods is mainly determined by the SNR of the image data [32]. While different SNR measures exist, here we define the noise level as the standard deviation of the intensities within the object, not the background. Correspondingly, the SNR is defined as the difference in intensity between the object and the background, divided by the standard deviation of the object noise [32]. Due to non-Gaussian noise statistics in the images, apart from linear Gaussian filtering [159] or wavelet-based denoising [108, 154], frequently nonlinear methods, such as median filtering [18] or anisotropic diffusion filtering [168] are used.

For studying intracellular dynamics, it may sometimes be necessary to compensate for the global motion of the cell, so as not to over- or underestimate local motion parameters. For this purpose, rigid or nonrigid image registration can be used [123, 150], which came to biological imaging mainly from medical image analysis, where it has been used on a regular basis for years. Another approach is to track the cells over time using existing cell tracking methods [43, 44, 192] and derive the final estimates and conclusions by combining both (cellular and intracellular) sources of information.

#### 1.3.2 Object Localization

After the preprocessing step, object detection methods are applied to the image data in order to locate objects of interest and accurately estimate their positions. The



simplest detection algorithms are based on image intensity thresholding with the underlying assumption that the real objects are brighter than background structures. In the image regions that indicate object presence after thresholding, the object position can be estimated using the centroid method [26, 32]. For a single axis in a 2D image  $I$ , the estimate is given by

$$x_c = \sum_x \sum_y xI(x, y) / \sum_x \sum_y I(x, y),$$

where  $I(x, y)$  is the image intensity value at position  $(x, y)$  and the summation is done over a small image region (mask) that contains the object. Depending on the mask size and image quality, in order to eliminate the bias in the estimated position towards the center of the mask, sometimes only positive differences between the intensity values and the threshold for each pixel within the mask are used in the centroid method [32]. For reasonable performance, such simple methods normally require images with relatively uniform background and high SNR, which makes them unsuitable for most live-cell imaging experiments.

More advanced detectors use additional features, such as object size, shape, volume, etc., for better discrimination from irrelevant background structures. By using additional features, these methods better model the object appearance and try to fit the models to the image data using some similarity measures. The model fitting is usually done by minimizing a predefined error measure (e.g. least squares fitting), or by measuring how good the model correlates with the data. The latter can be done, for example, by computing the normalized covariance for the small intensity template  $T$  that describes the object appearance and the original image  $I$ . This method is an extension of simple correlation with the template  $T$  [32], which originally cannot deal with nonuniform backgrounds. In 2D, the normalized covariance is given by<sup>1</sup>

$$C(x, y) = \sum_{i=-n}^n \sum_{j=-m}^m \frac{(I(x+i, y+j) - \bar{I}(x, y))(T(i+n, j+m) - \bar{T})}{M_I(x, y)M_T},$$

where  $T(i, j)$  is a  $(2n+1) \times (2m+1)$  intensity template,  $\bar{T}$  the mean value of the template intensity,  $\bar{I}(x, y)$  the mean value of the image intensity in the area overlapping with the template,  $M_T$  the variance of the template intensity, and  $M_I(x, y)$  is the variance of image intensity in the area overlapping with the template [32]. The local maxima in the resulting map  $C(x, y)$  indicate the image regions which are highly similar in appearance to the template. By applying a threshold to  $C(x, y)$ , these regions can be extracted and the object positions can be computed using the centroid method. The normalized covariance can cope with nonuniform background intensity and the only limiting assumption is the fixed and known shape of the searched object.

Another similarity measure that can be used to measure the correspondence between the object appearance template  $T$  and the spatial intensity distributions in the image data is a sum of absolute differences (SAD). In this method, the SAD map is

<sup>1</sup>The extension of this and subsequent formulae to 3D is straightforward.

computed for all possible shifts of the template  $T$  in the image as

$$\text{SAD}(x, y) = \sum_{i=-n}^n \sum_{j=-m}^m |I(x+i, y+j) - T(i+n, j+m)|.$$

The minima in the map  $\text{SAD}(x, y)$  correspond to the best fits. For multiple object detection the local minima are counted as found objects. Compared to covariance based detection, this method is highly sensitive to intensity scaling of the image and template, which can cause problems in practice since the fluorescent tags are bleaching during acquisition.

For the described correlation based methods, the accuracy of the position estimates is on the order of one pixel, since the shifts of the template are calculated on a discrete pixel grid. The accuracy of the object localization can be substantially increased by using detection methods that fit the object appearance model to the image data. Since the objects under consideration are smaller than the resolution of the imaging devices, the model of the PSF (for example the Gaussian approximation) can be used in order to model object appearance. For multiple object detection in 2D images, the fitting is performed in all the regions of the image where the probability of object existence is high, by minimizing the sum of squared differences

$$\text{MSE}_g(x, y, A, B) = \sum_i \sum_j \left( I(i, j) - A \exp \left( -\frac{(i-x)^2 + (j-y)^2}{2\sigma^2} \right) - B \right)^2.$$

The parameters that locally minimize the  $\text{MSE}_g(x, y, A, B)$ , are taken as the features of the found object [32, 161]. This approach is computationally expensive, but it demonstrates the highest accuracy in estimating the object position [32]. The latter conclusion comes from the study [32], where the approaches described above were quantitatively compared under different controlled conditions using artificial 2D time-lapse image sequences and is true only in the case of high SNR image data. For low SNR levels ( $< 5$ ), which are not uncommon in live-cell fluorescence microscopy imaging, the PSF model fitting breaks down [161].

### 1.3.3 Solving the Correspondence Problem

Once the objects have been detected in the image sequence, sets of estimated positions  $\{\{\mathbf{r}_t^k\}_{k=1}^{M_t}\}_{t=1}^{T_0}$  are available for the next processing step, where  $\mathbf{r}_t^k = (x_t^k, y_t^k, z_t^k)^T$  defines the position of object  $k$  in frame  $t$ ,  $M_t$  is a time varying number of objects per frame, and  $T_0$  is the number of frames in the image sequence. In order to obtain trajectories, the correspondence between the object positions in different frames needs to be established. Solving the correspondence problem is not a trivial task. In our application, the objects of interest are more or less identical and because of that searching for the corresponding objects in different frames on a basis of appearance information will not produce good results. In practice, the detection procedures are imperfect, which leads to spurious and missing objects that influence the accuracy of the linking procedure. Moreover, real objects can move densely together, be temporarily occluded and/or appear and disappear from the field of view during imaging.

Frequently, such ambiguous scenarios cannot be correctly dealt with using the existing tracking algorithms, and sometimes such situations confuse even expert biologists. This is especially the case when complicated, essentially 3D intracellular processes are studied using 2D confocal slicing.

In the case of almost indistinguishable objects, the linking procedure is mainly based on assumptions about the underlying object motion. The most frequently used motion models are the nearest-neighbor model (NNM) and the smooth motion model (SMM) [33, 171]. The NNM does not incorporate velocity information and is solely based on positional information. For object  $k$  in frame  $t$  and candidate object  $s$  in frame  $t + 1$ , the score  $c_t^{\text{NN}}(k, s)$  is defined as  $c_t^{\text{NN}}(k, s) = \|\mathbf{r}_t^k - \mathbf{r}_{t+1}^s\|$ . The object pair with the lowest score has the highest chance to be linked. If an object stays in one place, the score  $c_t^{\text{NN}}(k, s) = 0$ . The SMM, on the other hand, assumes that both velocity direction and magnitude change slowly from frame to frame. For this model, the corresponding score is defined as

$$c_t^{\text{SM}}(k, s) = w \left( 1 - \frac{\mathbf{v}_t^k \cdot \mathbf{v}_t^{ks}}{\|\mathbf{v}_t^k\| \|\mathbf{v}_t^{ks}\|} \right) + (1 - w) \left( 1 - \frac{2\sqrt{\|\mathbf{v}_t^k\| \|\mathbf{v}_t^{ks}\|}}{\|\mathbf{v}_t^k\| + \|\mathbf{v}_t^{ks}\|} \right),$$

where  $\mathbf{v}_t^k = \mathbf{r}_t^k - \mathbf{r}_{t-1}^k$ ,  $\mathbf{v}_t^{ks} = \mathbf{r}_{t+1}^s - \mathbf{r}_t^k$ , and  $w$  is a weighting coefficient. The first term in the expression for  $c_t^{\text{SM}}(k, s)$  accounts for the angular deviation of the displacement vectors by computing their dot product. The second term accounts for the speed deviation. Using this score, a candidate object in frame  $t + 1$  is searched that best satisfies the uniform motion assumption. If the object moves uniformly,  $\mathbf{r}_t^k - \mathbf{r}_{t-1}^k = \mathbf{r}_{t+1}^s - \mathbf{r}_t^k$ , and  $c_t^{\text{SM}}(k, s) = 0$ . If applicable, appearance similarity measures can be used in addition to the described spatial proximity criteria. Furthermore, the combination of these measures can be used to define some probability of assignment as a score, e.g.

$$c_t^{\text{P}}(k, s) = \exp \left( -(\mathbf{r}_{t+1}^s - \mathbf{r}_t^k)^T \mathbf{\Sigma} (\mathbf{r}_{t+1}^s - \mathbf{r}_t^k) \right) \exp \left( -\frac{(I_{t+1}^s - I_t^k)^2}{\sigma_I^2} \right),$$

where  $I_{t+1}^s$  and  $I_t^k$  are the intensities of objects  $s$  and  $k$  in the corresponding frames, and  $\mathbf{\Sigma}$  and  $\sigma_I^2$  are the parameters that account for small deviation in displacement and variation in intensity, respectively.

In order to link the objects and form the trajectories, the described assignment scores can be used in several ways. First, there are greedy algorithms that make decisions about the best assignment by taking into account the score values only in the current frame. The disadvantage of the greedy search is its tendency to stop in the first local minimum of the searched space. At the same time, if the density of the objects in the image data is relatively low, and the motion is either slow or uniform, so that the NNM or SMM are appropriate, then the greedy approach is a good choice (also because it is computationally quite cheap). In general, the linking procedures can operate either globally in space, where the assignment is performed jointly for all objects in one frame, or globally in time, depending on how many frames are taken into account at the same time for similarity measurement, or both (be global in space and time). Most of the time, a greedy assignment is done first. Then, the iterative

procedures start to make changes in these assignments and check how the global score behaves (if it is lowered).

Many linking techniques solve the correspondence problem in a spatially global manner, by defining a global score that is afterwards minimized [33, 52, 74, 132, 166, 171]. For example, some form of global optimization can be accomplished using graph theory [132]. This approach is implemented in a publicly available tracking software *ParticleTracker*, the performance of which is evaluated in Chapter 3. Here, linking is based on finding the spatially global solution to the correspondence problem in a given number of successive frames. The solution is obtained using graph theory and global energy minimization [132]. The linking also utilizes the zero- and second-order intensity moments of the object intensities, which helps to resolve object intersection problems and improves the linking results.

Another solution to the correspondence problem can be obtained by using dynamic programming [128]. With dynamic programming, the total cost, which is in this case the weighted sum of  $c_t^{\text{NN}}$  and object intensity  $I_t$ , is optimally minimized in a temporally global way. With this approach, tracking of a single object through the entire image sequence is possible [128]. Multiple object tracking can be achieved by tracking the objects one by one, which is not an attractive and workable solution for image data with large numbers of interacting objects.

Recently presented advanced linking techniques use fuzzy-logic and linear assignment problem (LAP) frameworks. In the former approach [74], four cost functions that measure the object similarity in consecutive frames are introduced: two of them are similar to the two summands in  $c_t^{\text{SM}}$ , and two additional costs are based on the objects appearance,  $c_t^I(k, s) = 1 - |I_t^k - I_{t+1}^s| / |I_t^k + I_{t+1}^s|$  and  $c_t^S(k, s) = 1 - |A_t^k - A_{t+1}^s| / |A_t^k + A_{t+1}^s|$ , where  $I_t^k$  is the total intensity and  $A_t^k$  is the total area of the spot  $k$  in frame  $t$ . Further, the fuzzy-logic system is employed to estimate the similarity between the object in frame  $t$  (parent object) and a set of candidate objects in frame  $t + 1$ . Fuzzy logic is a form of multi-valued logic derived from fuzzy set theory to deal with reasoning that is approximate rather than precise. A set of if-then rules is introduced, where each rule uses the values of the four similarity measures and outputs a real value between 0 and 1. This gives the possibility to extend the binary concept that a parent object is similar (“1”) or not similar (“0”) to a candidate object to a broader range: “least-similar”, “median-similar”, “most-similar”, etc. The outputs of all the rules are aggregated and a common score is derived for each candidate object. The parent object is connected with the candidate object that has the highest score. With this approach, fuzzy rule selection plays an important role and it strongly affects the performance of the tracking algorithm. Additionally, in the described algorithm [74], the linking is performed separately for each object, so the whole procedure is global neither in time, nor in space.

One of the most recent approaches in the literature [72] constructs the set of trajectories from the set of detected objects in two steps. First, the greedy assignment between the consecutive frames is performed using the cost function based on the distance between two objects. This step produces many short and broken tracks. The second step attempts to link the track segments (close the gaps) and deal with track splitting and merging using additionally the object intensity information. For this stage, corresponding closing, splitting, and merging cost functions are defined, which

have to be tailored to the specific application. Although the first step is greedy, solving the subsequent track segment optimization problem is done globally, overcoming the shortcomings of the previously described techniques. The method was shown to perform robust tracking of multiple objects under high-density conditions [72]. The shortcoming of this and the previously described method is again the separation of the detection and tracking procedures.

A somewhat different solution to the correspondence problem is presented in [17]. Rather than adopting the usual frame-by-frame approach, the authors consider the time-lapse 2D+t image sequence as one 3D spatiotemporal volume, where the tracks appear as 3D curves. The correspondence problem is then solved by finding the geodesics in a Riemannian metric computed from the 3D image. Similarly to the method described in [72], the cost optimization procedure, which is global only in time, is split into two steps. After the object detection, the nearby objects are grouped into short trajectories that are not complete due to possibly poor detection results. Then, partial tracks are linked with minimal paths to constitute complete tracks. The construction of minimal paths takes into account information from both image features and tracking constraints (maximum object displacement, etc.). Moreover, each time a minimal path is added to a trajectory, image information is removed along the path in order to avoid trajectories to merge.

The quality of the solution to the correspondence problem highly depends on the nature of the dynamical processes that were studied in the experiments, for example the number of objects, density of objects, type of motion, etc. Many of the described methods perform poorly when applied to biological data because of too simplistic assumptions of object behavior, which cannot cope with the real heterogeneity of subcellular dynamics. Additionally, due to separation of the tracking procedure into detection and linking, for low quality image data, the linking methods have to deal with lots of spurious objects detected in the first stage. Commonly, detectors do not specify any confidence measure for each detected object, that could be used to distinguish real objects from possible false detections. If that would be possible, the results of linking could be improved. Such confidence measure, for instance, can be specified in terms of variance in the object position measurements and is frequently used in probabilistic tracking approaches, which are the focus of this thesis.

#### 1.3.4 Probabilistic Methods for Tracking

Solving the correspondence problem and creating tracks can also be described as a state estimation problem and solved using probabilistic methods [9, 126]. Probabilistic tracking is a state estimation problem, where the object hidden state  $\mathbf{x}_t$  is estimated in time based on previous states, noisy measurements  $\mathbf{z}_t$ , and prior knowledge about object properties. Mathematically, it can be formulated as

$$\mathbf{x}_t = f_t(\mathbf{x}_{t-1}, \mathbf{v}_t), \quad \mathbf{z}_t = h_t(\mathbf{x}_t, \mathbf{u}_t), \quad (1.1)$$

where  $f_t$  and  $g_t$  are possibly nonlinear state transition and observation models respectively, and  $\mathbf{v}_t$  and  $\mathbf{u}_t$  are white noise sources. If the measurement-to-object association is known, (1.1) can be solved either exactly (when  $f_t$  and  $g_t$  are linear and  $\mathbf{v}_t$  and  $\mathbf{u}_t$

are Gaussian) using the Kalman filter, or (in the general case) using SMC approximation methods [9]. The solution is the posterior probability distribution function (pdf)  $p(\mathbf{x}_t|\mathbf{z}_{1:t})$ , where  $\mathbf{z}_{1:t} = \{\mathbf{z}_1, \dots, \mathbf{z}_t\}$ , from which minimum mean square error (MMSE) or maximum a posteriori (MAP) state estimations can be easily computed [9].

In order to obtain the trajectory estimate for one object using the Kalman filter, the state vector  $\mathbf{x}_t$ , which may include object position, velocity, acceleration, etc., and which cannot be directly measured is estimated on the basis of noisy measurements  $\mathbf{z}_{1:t}$ , for example extracted positions  $\mathbf{r}_{1:t}$  using detection methods described above. It is assumed that the state transition and the observation process are specified as follows,

$$\mathbf{x}_t = F_t \mathbf{x}_{t-1} + \mathbf{v}_t, \quad \mathbf{z}_t = H_t \mathbf{x}_t + \mathbf{u}_t, \quad (1.2)$$

where  $F_t$  and  $H_t$  are system matrices defining the linear functions, and the covariances of  $\mathbf{v}_t$  and  $\mathbf{u}_t$ , which are statistically independent random variables with zero mean, are respectively  $Q_t$  and  $R_t$ . The solution of (1.2),  $p(\mathbf{x}_t|\mathbf{z}_{1:t})$ , in this case is given by the following recursive relationship:

$$\begin{aligned} p(\mathbf{x}_{t-1}|\mathbf{z}_{1:t-1}) &= \mathcal{N}(\mathbf{x}_{t-1}|\mathbf{m}_{t-1|t-1}, P_{t-1|t-1}), \\ p(\mathbf{x}_t|\mathbf{z}_{1:t-1}) &= \mathcal{N}(\mathbf{x}_t|\mathbf{m}_{t|t-1}, P_{t|t-1}), \\ p(\mathbf{x}_t|\mathbf{z}_{1:t}) &= \mathcal{N}(\mathbf{x}_t|\mathbf{m}_{t|t}, P_{t|t}), \end{aligned} \quad (1.3)$$

where

$$\begin{aligned} \mathbf{m}_{t|t-1} &= F_t \mathbf{m}_{t-1|t-1}, \\ P_{t|t-1} &= Q_{t-1} + F_t P_{t-1|t-1} F_t^T, \\ \mathbf{m}_{t|t} &= \mathbf{m}_{t|t-1} + K_t (\mathbf{r}_t - H_t \mathbf{m}_{t|t-1}), \\ P_{t|t} &= P_{t|t-1} - K_t H_t P_{t|t-1}, \end{aligned} \quad (1.4)$$

and where  $\mathcal{N}(\cdot|\mathbf{m}, P)$  is a Gaussian distribution with mean  $\mathbf{m}$  and covariance  $P$ , and

$$\begin{aligned} S_t &= H_t P_{t|t-1} H_t^T + R_t, \\ K_t &= P_{t|t-1} H_t^T S_t^{-1}. \end{aligned}$$

For multiple object tracking the same framework can be used, but the tracking in this case is complicated by the ambiguous measurement-to-object associations – for every measurement given by the detector at time  $t$  it is necessary to know which object it has to be used for to update the predicted state in (1.4). In practice that information is not available. The most efficient tracking approaches that are able to deal with such missing information and still perform tracking, are the multiple hypothesis tracker (MHT) and the joint probabilistic data association (JPDA) filter [15]. The former builds a tree of hypotheses about all possible measurement-to-track associations, and because of that is not suitable for tracking large numbers of objects. The standard JPDA filter is designed for linear Gaussian models in (1.1) and uses all measurements to update each track estimate [15]. For practical reasons, measurement gating is often used, which selects for each object the subset of measurements that most likely originated from the object. Contrary to applications where sensors provide information about the number of objects and their positions, JPDA cannot be

applied directly to our applications, because actual position or velocity measurements are not available, but need to be derived from the image data first.

For analysis of subcellular dynamics a few approaches have been proposed that implement the described probabilistic framework [52, 146]. The first one extends the JPDA filter by using the  $h$ -dome detector (see also Chapter 2) and is shown to perform accurate and robust tracking of microtubules, which are growing with almost constant velocity. The second approach [52] implements the idea of interacting multiple model (IMM) filtering [11], which was initially designed only for linear Gaussian models. This type of filtering is useful when it is necessary to track objects that exhibit different types of motion patterns in the same image sequence. Here, several motion models  $F_t$  are employed, which predict the object position from frame to frame using the Kalman filter. The method was shown to perform extremely well in comparison with standard Kalman filtering for tracking of endocytosed quantum dots. In Chapter 4, a PF-based method is developed that generalizes the idea of IMM.

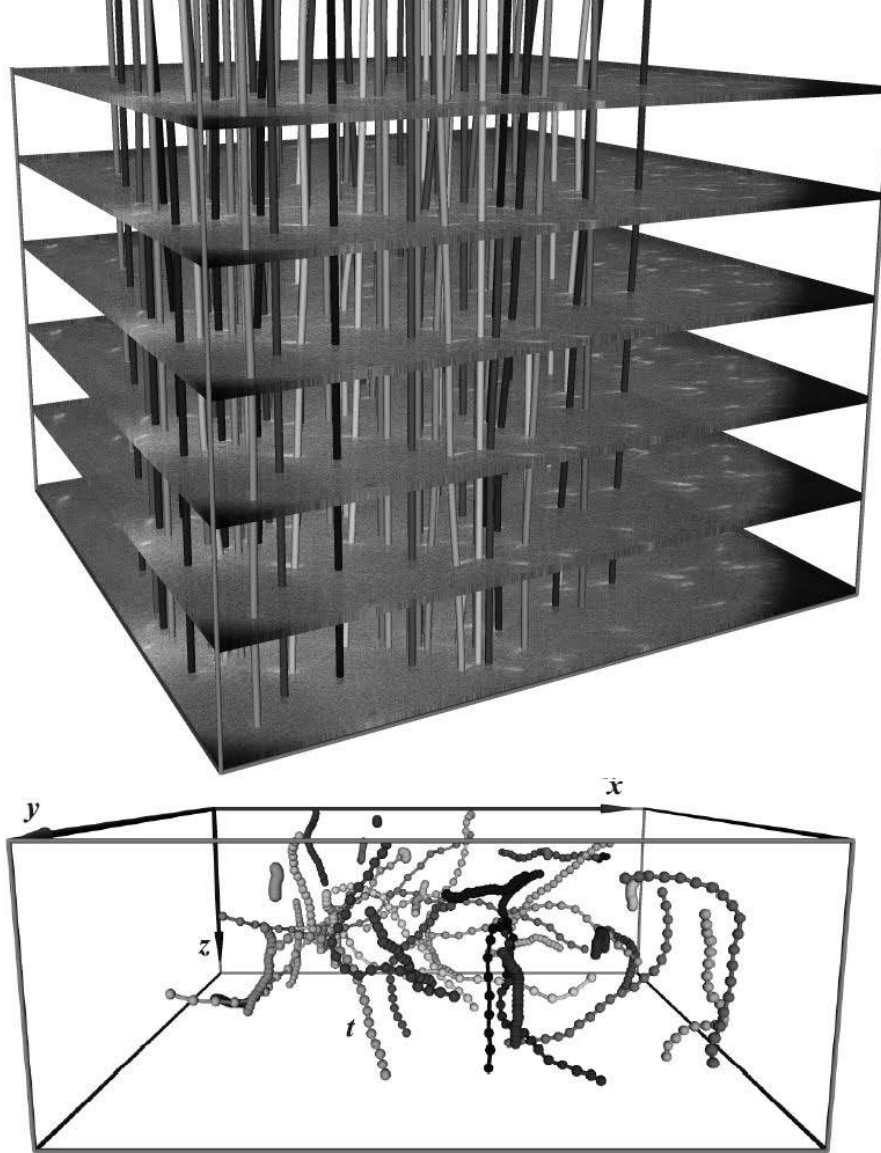
With the probabilistic tracking approaches, especially in the case of multiple object tracking, it is also beneficial to specify any prior knowledge about object interactions, additionally to the modeling of the object dynamics. Tracking approaches that assume a one-to-one measurement-to-track assignment (as in most of the deterministic tracking approaches and some of the probabilistic ones), fail to resolve the most ambiguous track interaction scenarios, where two or more objects come in close proximity to each other and produce only one measurement for a few time frames. By incorporating prior knowledge about the objects to be tracked (for example, microtubules are rigid structures that cannot easily bend, and because of that their direction of movement before and after the interaction should be approximately the same), the rate of incorrectly switched tracks can be greatly reduced [141, 146].

## 1.4 Analyzing Tracking Results

In time-lapse microscopy, the final step of the analysis consists of interpretation of the detection and tracking results in order to confirm or reject hypotheses that were tested during the experiment, or qualitatively or quantitatively look for new findings that would lead to new hypotheses and correspondingly to new experiments. Before applying any quantitative techniques, some qualitative verification of the obtained tracking results might be useful. This is especially true for low SNR image data, where lots of automated techniques either break down or produce nonsensical tracks. For this purpose, modern computer graphics rendering and visualization techniques can be used (see an example in Fig. 1.2) so as to assist in the verification of the tracking and give some initial impression about the possible trends in the data and which quantitative methods for the analysis to choose [54].

Once the results of tracking are verified, a multitude of measures about the geometry of the trajectories and additionally the object appearance can be readily obtained. An example is the total distance traveled by the object or the mean square displacement, which are typically used to study the diffusion characteristics of the motion of individual objects [8, 122, 128, 131, 158]. Other commonly studied parameters are average and instantaneous velocities. Instantaneous object velocity is estimated as





**Figure 1.2.** Different visualizations of time-lapse image data: combined visualizations of image frames and tracks giving a qualitative impression of the accuracy and consistency of the tracking results (top), and spatiotemporal view of tracks from an artificial 3D time-lapse image sequence (data not shown), with the time coordinate indicated along the trajectories by small spheres (bottom).

the distance traveled by the object between two consecutive frames divided by the corresponding time interval. Average velocity is computed as the sum of the frame-



to-frame distances traveled, divided by the total time elapsed. In many experiments it is enough to estimate the average velocity either per track or a number of tracks (and the variance of those estimates) in order to derive some conclusions about the hypotheses being tested. On the other hand, with current advanced imaging techniques, biologists are eager to inspect and analyze the intracellular motion in more detail. This can be achieved by studying the instantaneous velocities and their distributions [92, 167–169]. Contrary to the average velocity estimate, histograms of instantaneous velocities provide insight into the possible heterogeneity of the intracellular motion and reveal the most dominant modes of motion. Additionally, object acceleration can be easily estimated, but is rarely studied.

Ideally, automated tracking techniques in molecular biology should facilitate the study of behavioral heterogeneity, to find and classify distinctive motion patterns (or confirm absence of such) depending on the experimental conditions. Knowing the typical behavior patterns of “healthy” molecular processes, it will be much easier to understand abnormal behavior that leads to disease and to define strategies that return the deviated system to its normal state. Therefore, comprehensive and automated analysis of large scale experimental data is especially important.

## 1.5 Thesis Outline

The subject of this thesis is tracking of multiple subcellular objects using time-lapse microscopy imaging. The main focus is on the development of robust and accurate automatic tracking algorithms, built within a probabilistic framework. The Bayesian tracking framework, which recently has become popular in other research fields and was shown to outperform deterministic methods, is capable of solving complex estimation problems by combining available noisy measurements (images, extracted object positions, etc.) with prior knowledge about the underlying object dynamics and the measurement formation process. Nevertheless, it is still only a framework, which gives the solution in a very general form, independent of applications. In order to apply the Bayesian approach in practice, the “ingredients” of the framework must be made application specific. In our case, these are the image formation process, defining the object appearance in the images, the noise sources that influence the image quality, and prior knowledge about the object behavior. The more accurate these aspects are specified and modeled, the closer the estimation to optimal. Nevertheless, even with all these ingredients in place, the optimal Bayesian solution is analytically tractable only in a restrictive set of cases – for example, the Kalman filter provides an optimal solution in the case of linear dynamic systems with Gaussian noise. For most practical cases, approximation techniques must be used. One of the most powerful and especially suitable for this purpose are sequential Monte Carlo methods, also known as particle filtering<sup>2</sup> (PF). With that in mind, this thesis describes a set of PF-based methods, that have been developed and evaluated for tracking of multiple objects in a variety of time-lapse biological studies.

<sup>2</sup>In this thesis (except Chapter 2), the word “particle” does not refer to any real subcellular structures and, because of possible confusion, everywhere in the thesis the word “object” is used for those structures. The word “particle” is reserved for the PF methods.

The thesis organized as follows. First, in Chapter 2, a quantitative comparison of frequently used object detection methods applied to fluorescence microscopy images is described. Even in the case of probabilistic tracking methods, object detection methods are useful, for example for detection of appearing and disappearing objects during tracking. In the chapter, six supervised and two unsupervised (machine learning) techniques are quantitatively evaluated and compared, using both synthetic image data and images from real biological studies. A comparison of this sort has not been carried out before in the literature. Next, in Chapter 3, a new PF-based tracking technique is proposed for tracking of subcellular structures moving with nearly constant velocity, such as microtubules. Experiments on synthetic as well as real fluorescence microscopy image sequences demonstrate the superior performance of the new method compared to popular frame-by-frame tracking methods. Chapter 4 presents an extension of the method developed in Chapter 3, which is able to track multiple types of intracellular objects (microtubules, vesicles, and androgen receptors) and can deal with different types of motion patterns. For that, several improvements over the previous PF are developed. Finally, Chapter 5 describes another biological application, where microtubule dynamics is studied *in vitro*. It presents a novel PF-based approach for analysis of the image data and estimation of important microtubule dynamics parameters. For this application, a special type of particle filters is designed, for the tracking of spatiotemporal structures. The results presented in the various chapters lead to the general conclusion that PF-based methods are very suitable for subcellular object tracking in biological microscopy and are superior to existing deterministic approaches.

# Quantitative Comparison of Spot Detection Methods in Fluorescence Microscopy

---

*Not everything that can be counted counts, and not everything  
that counts can be counted.*

— ALBERT EINSTEIN (1879-1955)

**Abstract** — Quantitative analysis of biological image data generally involves the detection of many subresolution spots. Especially in live cell imaging, for which fluorescence microscopy is often used, the signal-to-noise ratio (SNR) can be extremely low, making automated spot detection a very challenging task. In the past, many methods have been proposed to perform this task, but a thorough quantitative evaluation and comparison of these methods is lacking in the literature. In this chapter, we evaluate the performance of the most frequently used detection methods for this purpose. These include six unsupervised and two supervised methods. We perform experiments on synthetic images of three different types, for which the ground truth was available, as well as on real image data sets acquired for two different biological studies, for which we obtained expert manual annotations to compare with. The results from both types of experiments suggest that for very low SNRs ( $\approx 2$ ), the supervised (machine learning) methods perform best overall. Of the unsupervised methods, the detector based on the so-called *h*-dome transform from mathematical morphology performs comparably, and has the advantage that it does not require a cumbersome learning stage. At high SNRs ( $> 5$ ), the difference in performance of all considered detectors becomes negligible.

## 2.1 Introduction

The very first stage in the analysis of biological image data generally deals with the detection of objects of interest. In fluorescence microscopy, which is one of the most basic tools used in biology for the visualization of subcellular components and their dynamics [88, 100, 113, 156, 164, 180], the objects are labeled with fluorescent proteins and appear in the images as bright spots, each occupying only a few pixels (see Fig. 2.1 for sample images). Digital image analysis provides numerical data to quantify and substantiate biological processes observed by fluorescence microscopy [3, 45, 97, 185, 191]. Such automated analysis is especially valuable for high-throughput imaging in proteomics, functional genomics and drug screening [42, 103]. Nevertheless, obtaining accurate and complete measurements from the image data is still a great challenge [38]. In many cases, the quality of the image data is rather low, due to limitations in the image acquisition process. This is especially true in live cell imaging, where illumination intensities are reduced to a minimum to prevent photobleaching and photodamage, resulting in a very low signal-to-noise ratio (SNR) [53, 95, 96]. In addition, despite recent advances in improving optical microscopy [51, 63], the resolution of even the best microscopes available today is still rather coarse (on the order of 100 nm) compared to the size of subcellular structures (typically only several nanometers in diameter), resulting in diffraction-limited appearance. As a consequence, it is often difficult, even for expert biologists, to distinguish objects from irrelevant background structures or noise.

In practice, automated object detection methods applied to fluorescence microscopy images either report too many false positives, thus corrupting the analysis with the presence of nonexistent objects, or they detect less objects than are in fact present, causing subsequent analyses to be biased towards more clearly distinguishable objects. This is also a serious issue in time-lapse imaging, where the objects of interest are to be tracked over time to study their dynamics. In common tracking algorithms, which consist of separate detection (spatial) and linking (temporal) stages [95, 96], the performance of the detector is crucial: poor detection likely causes the linking procedure to yield nonsensical tracks, where correctly detected objects in one frame are connected with false detections in the next (and vice versa), or where tracks are terminated prematurely because no corresponding objects were detected in the next frame(s). Modern tracking approaches, based on Bayesian estimation [141, 142], avoid the hard decision thresholds in the detection stage of conventional approaches, and describe object existence in terms of probability distribution functions (pdf). Such real-valued pdfs reflect the degree of believe in the presence of an object at any position in the image in a more “continuous” fashion, in contrast with the binary representation (either “present” or “not present”) obtained after applying hard thresholds. Nevertheless, even in probabilistic tracking frameworks, some form of “deterministic” object detection is still necessary in the track initiation and termination procedures [141, 142, 146], again illustrating the relevance of having a good spot detector. Several detectors have been proposed in the literature, and the classic, relatively simpler methods have been compared previously for tracking [26, 32], but a thorough quantitative comparison including recent, more complex methods is missing.

In this study, we compare several detectors that are frequently used for object

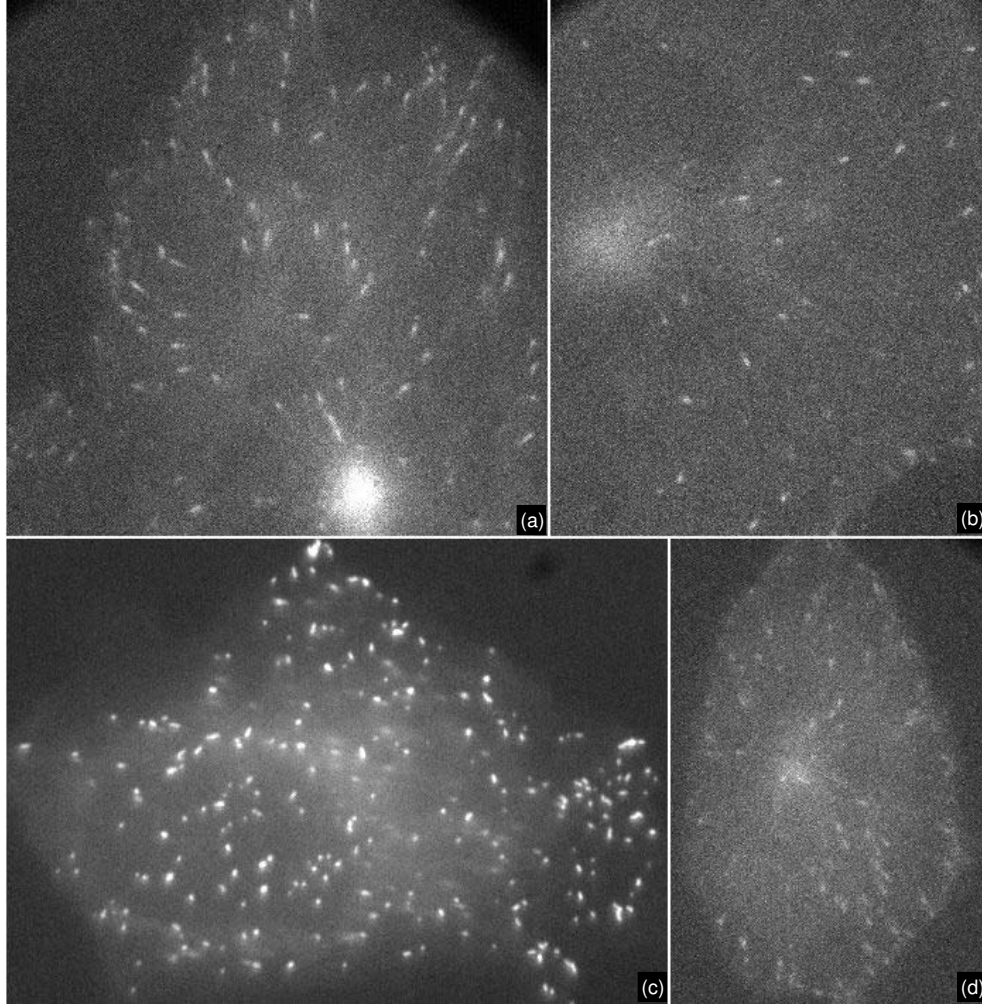
detection in fluorescence microscopy imaging, and quantify their performance using both synthetic images and real image data from different biological studies. The sensitivity of the methods is studied as a function of their parameters and image quality (expressed in terms of SNR). The methods under consideration range from relatively simple local background subtraction [185], to linear or morphological image filtering [20, 21, 128, 142, 146, 161], to wavelet-based multiscale products [52, 108], and machine learning methods [73]. They can be divided into two groups: unsupervised and supervised. The first consists of algorithms that (implicitly or explicitly) assume some object appearance model and contain parameters that need to be adjusted either manually or semi-automatically in order to get the best performance for a specific application. Supervised methods, on the other hand, “learn” the object appearance from annotated training data—usually a large number of small image patches containing only the object intensity profiles (positive samples) or irrelevant background structures (negative samples).

This chapter is organized as follows. First, in Section 2.2, we provide background information on the image formation process in fluorescence microscopy and describe the object detection framework in general. This helps to put the different detection methods in proper perspective and provides motivations for some of the choices made later on in the chapter. The detection methods that were considered in this study and that implement the general framework are described in Section 2.3. Next, in Section 2.4, we present the experimental results of applying the detection methods to synthetic images, for which ground truth was available, as well as to real fluorescence microscopy image data from several biological studies. A concluding discussion of the main findings and their implications is given in Section 2.5.

## 2.2 Detection Framework for Fluorescence Microscopy

### 2.2.1 Image formation

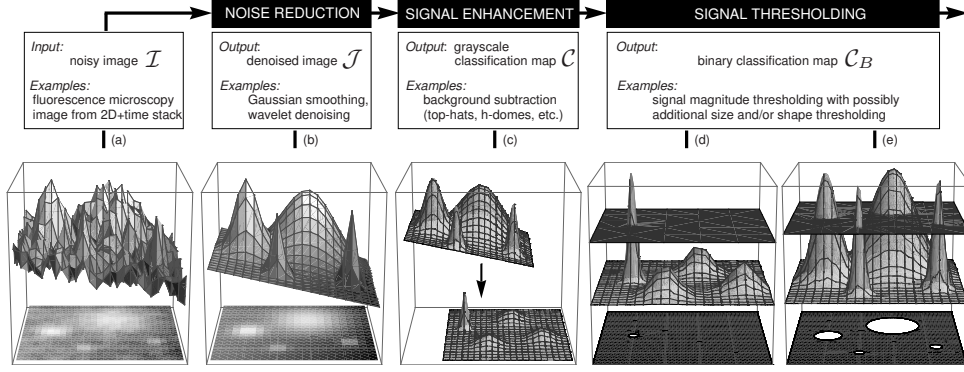
In fluorescence microscopy, specimens are labeled with fluorophores. The distribution of fluorescence caused by exciting illumination is then observed and captured by a photosensitive detector (usually a CCD camera or a photomultiplier tube) that measures the intensity of the emitted light and creates a digital image of the sample. The objects of interest in our application appear in images as blurred spots, which are relatively small and compact, have no clear borders (which is why we prefer to speak of “detection” rather than “segmentation”), and their intensity is higher than the background. The blurring is caused by the diffraction phenomenon and imperfections of the optical system, which for commonly used confocal microscopes limits the resolution to about 200 nm laterally and 600 nm axially [95, 161, 185, 190]. This is characterized by the point spread function (PSF) of the system, which is the image of a point source of light. In our applications, the theoretical PSF, which can be expressed by the scalar Debye diffraction integral [190], can in practice be approximated by a 2D or 3D Gaussian PSF [161], depending on the dimensionality of the image data.



**Figure 2.1.** Sample images of microtubules (a,b,d) and peroxisomes (c) labeled with green fluorescent protein (GFP) and imaged using confocal microscopy. The images are single frames from 2D time-lapse studies, acquired under different experimental conditions. The quality of the images ranges from  $\text{SNR} \approx 4\text{--}6$  (a,c) to  $\approx 2\text{--}4$  (b,d).

Apart from the diffraction-limited spatial resolution, another major source of aberrations introduced in the imaging process is intrinsic photon noise, which results from the random nature of photon emission. Photon noise, which is independent of the detector electronics, can be reduced (and, consequently, the SNR increased) only by increasing the light intensity or the exposure time. However, increasing the light intensity in order to improve the image quality causes the fluorescent signal to fade permanently due to photon-induced chemical damage and covalent modification,





**Figure 2.2.** Object detection framework. The original noisy image (a) is preprocessed with some noise reduction method, and the resulting image (b) is transformed (enhanced) into a new image (c), in which the possible object locations have higher signal magnitude than all other structures (d), or all the suspicious locations are marked (e). The threshold (represented by the dark-gray planes in (d) and (e)) is applied and the connected components in the binarized image (white clusters on the black background) are counted as the detected objects.

a process called photobleaching [185]. While this effect can be exploited to study specific dynamical properties of particle distributions [87, 156], it hampers detection and tracking of individual fluorescent particles. With a laser as excitation source, photobleaching is observed on the time scale of microseconds to seconds, and should be taken care of especially in time-lapse microscopy.

In this study, we deal with subresolution objects (blurred spots) on a possibly nonuniform background, the appearance of which can be modeled using a Gaussian approximation of the PSF. While for experimental and illustration purposes we limit ourselves to 2D image data, all detection methods considered in this chapter can be applied straightforwardly to 3D data without any substantial changes. Each image  $\mathcal{I}$  consist of  $N_x \times N_y$  pixels, where each pixel corresponds to a rectangular area of dimension  $\Delta_x \times \Delta_y \text{ nm}^2$  and the measured intensity at position  $(i, j)$  is denoted as  $I(i, j)$ . In other words  $\mathcal{I} = \{I(i, j) : i = 1, \dots, N_x, j = 1, \dots, N_y\}$ . In order to model different types of subcellular particles (round or elongated appearance), we use an asymmetric 2D Gaussian function. In this case, the measured intensity at  $(i, j)$  caused by the fluorescent light source located at  $(x, y)$ , which is the real-valued position within the image, is given by

$$I(i, j) = B(i, j) + \exp\left(-\frac{1}{2}\mathbf{m}^T \mathbf{R}^T \mathbf{\Sigma}^{-1} \mathbf{R} \mathbf{m}\right), \quad (2.1)$$

where  $\mathbf{\Sigma} = \text{diag}[\sigma_{\max}^2, \sigma_{\min}^2]$ ,  $\mathbf{R} = \mathbf{R}(\phi)$  is a rotation matrix

$$\mathbf{R}(\phi) = \begin{pmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{pmatrix}, \quad \mathbf{m} = \begin{pmatrix} i\Delta_x - x \\ j\Delta_y - y \end{pmatrix},$$

and  $-\pi < \phi \leq \pi$  defines the rotation,  $B(i, j)$  is the background intensity distribution, and the parameters  $\sigma_{\max}$  and  $\sigma_{\min}$  represent the blurring induced by the PSF and, at the same time, model the elongation of the object. For symmetrical subresolution structures such as vesicles,  $\sigma_{\min} = \sigma_{\max} \approx 80\text{--}100$  nm, and for the elongated objects, such as microtubules,  $\sigma_{\min} \approx 80\text{--}100$  nm and  $\sigma_{\max} \approx 250\text{--}300$  nm [141, 161]. Concerning the density of objects in our applications, typical  $512 \times 512$ -pixel images contain around 50–200 objects.

### 2.2.2 Detection Framework

Before we describe the different detection approaches evaluated in this chapter, we first consider the detection framework in general (Fig. 2.2). This framework can be split into three subsequent steps. Each detector considered in this chapter includes these steps, but may implement them in a different way. In practice, some of the steps are optional or can be combined. Taking as input the noisy images containing the objects of interest, possibly embedded in a nonuniform background (Fig. 2.2(a)), the detector proceeds as follows:

*Step 1 (Noise Reduction):* The input image  $\mathcal{I}$  is preprocessed using noise reduction techniques. In most cases, Gaussian smoothing [159] or matched filtering [165] is used, which may increase the SNR and improve image quality and object visibility. The output of this step is a filtered image  $\mathcal{J}$  (Fig. 2.2(b)).

*Step 2 (Signal Enhancement):* In this step, signal processing techniques are used that enhance the denoised fluorescent light signal *only* in those regions of the image  $\mathcal{J}$  where the actual objects are and, at the same time, suppress the light signal from all the background structures. That is, the image  $\mathcal{J}$  is transformed to a new grayscale image  $\mathcal{C}$  (Fig. 2.2(c)), also called here the grayscale classification map, which does not necessarily represent the object intensity distribution anymore. At this stage, the image  $\mathcal{C}$  is rather a 2D (or 3D) signal, the value of which at any pixel measures the certainty in the object presence at that position. In other words, the image  $\mathcal{C}$  can also be considered a probability map that describes possible object locations. Two examples of this classification map are shown in Fig. 2.2(d) and Fig. 2.2(e), where the image  $\mathcal{C}$  in Fig. 2.2(d) is the result of applying a correlation based technique (in this case a matched filter), which convolves the image  $\mathcal{J}$  with a PSF-like kernel and produces a high response in regions where objects are present (where the image intensity distribution matches the kernel), and a low response in all other image regions, suppressing the background structures. The image  $\mathcal{C}$  in Fig. 2.2(e) corresponds to the situation where local background subtraction is used based on the  $h$ -dome transformation [177], which “cuts off” the local maxima in the image  $\mathcal{J}$  in the dome-like shape of equal heights.

The described feature enhancement step does not actually detect features or objects. At this stage no quantitative information (about the object presence, its position, size, etc.) can yet be extracted and it is still up to the observer to visually link pixels that belong to one object.

*Step 3 (Signal Thresholding):* To obtain the number of objects and extract position information from the grayscale classification map, hard (binary) decision thresholds need to be applied. First, the image  $\mathcal{C}$  is thresholded, where the threshold  $l_d$  is



applied to the signal magnitude and the binary map  $\mathcal{C}_B$  is obtained (Fig. 2.2(d,e)). Disjoint clusters of connected nonzero pixels in  $\mathcal{C}_B$  correspond to detected objects and can be used to label the pixels in the original image  $\mathcal{I}$  for subsequent analysis of the object intensity distribution. Depending on the image  $\mathcal{C}$ , the result of thresholding may be sensitive to the value of  $l_d$ . In that case, a second threshold  $v_d = (v_{\min}, v_{\max})$  may be applied to the size and/or shape of the clusters: only those clusters in  $\mathcal{C}_B$  with size larger than  $v_{\min}$  and smaller than  $v_{\max}$  are labeled as detected objects.

In practice, the signal thresholding with  $l_d$  does not always produce fully connected regions (clusters of pixels) in  $\mathcal{C}_B$ , in places where the true objects are located. In most cases, because the noise is not completely removed during Step 1, clusters of nonzero pixels in  $\mathcal{C}_B$  that belong to the same spot are not connected or contain erroneous zero-pixels inside the cluster. In order to solve this problem, the closing operation from mathematical morphology [138, 149, 185] is frequently used as a post-processing step.

## 2.3 Detection Methods

In this section we describe the detection methods that were included in our study. All of them implement the three main steps of the general detection framework presented in the previous section. Some of the methods require noise reduction as an explicit preprocessing step to improve the detection performance, and in our analysis we include two techniques for this purpose (Gaussian filtering and wavelet denoising) that are computationally fast, easy to implement, and which are frequently used in practice (Section 2.3.1). The most characteristic feature of any detection method is its implementation of the second step of the framework (signal enhancement). As pointed out in the introduction, we make a distinction between unsupervised (Section 2.3.2) and supervised (Section 2.3.3) detection techniques. Some of them inherently reduce noise and thus do not require an explicit noise reduction step. The third step (signal thresholding) determines the final outcome of the detector, which is used to assess its performance. In the last subsection (Section 2.3.4) we describe how performance was measured in our study.

### 2.3.1 Noise Reduction

#### 2.3.1.1 Gaussian Smoothing

Noise reduction in this case consists of smoothing the original image  $\mathcal{I}$  with the Gaussian kernel  $G_\sigma$  at scale  $\sigma$ . The filtered image  $\mathcal{J}$  is obtained as

$$J(i, j) = (G_\sigma * I)(i, j) = \sum_{i'=1}^{N_x} \sum_{j'=1}^{N_y} G_\sigma(i - i', j - j') I(i', j'), \quad (2.2)$$

where  $*$  denotes the convolution operation. (Here, and in the rest of the chapter, for all methods that require the convolution of an image with a filter kernel or mask, the image is mirrored at the borders.) In the case of additive uncorrelated noise, this smoothing can be related to matched filtering [165], which maximizes the SNR in

the filtered images. This is because the PSF, which models the appearance (intensity profile) of the subcellular objects, can be approximated to a high degree of accuracy by a Gaussian [190]. The smoothed image  $\mathcal{J}$  can also be used as the grayscale classification map  $\mathcal{C}$ , due to the fact that the image  $\mathcal{J}$  is a correlation map that shows where objects similar in shape to the PSF are located. The object locations can be extracted by thresholding the image  $\mathcal{J}$  in Step 3 (see Fig. 2.2), but this approach does not work in practice for typical images, which usually contain inhomogeneous backgrounds and varying object intensities.

### 2.3.1.2 Isotropic Undecimated Wavelet Denoising

This wavelet-based filtering technique is frequently used for image denoising in different applications [152], but also for building a separate detection procedure (Section 2.3.2.1) [52,108]. The isotropic undecimated wavelet transform (IUWT) [152,154] is well adapted to the analysis of images which contain isotropic sources, such as in astronomy [154] or in biology [52,108], where the object appearance or shape is diffuse (no clear edges) and more or less symmetric. The denoising is accomplished by modifying the relevant wavelet coefficients and inverse transforming the result. The IUWT is usually favored over orthogonal discrete wavelet transforms (DWT) for this purpose [91]. Contrary to the DWT, the IUWT is redundant, but translation invariant, and the wavelet coefficient thresholding using an undecimated transform rather than a decimated one normally improves the result in denoising applications [151].

We used the B3-spline version of the separable 2D IUWT [108,152], which decomposes the original image into  $K$  wavelet planes (detail images) and a smoothed image, all of the same size as the original image. The image  $\mathcal{I}$  is first convolved row by row and then column by column with the 1D kernel  $[1/16, 1/4, 3/8, 1/4, 1/16]$ , which is modified depending on the scale  $k$  by inserting  $2^{k-1}$  zeros between every two taps. The image  $I_{k-1}(i, j)$  is convolved with the kernel giving a smoothed image  $I_k(i, j)$ , and the wavelet plane is computed from these two images as

$$W_k(i, j) = I_{k-1}(i, j) - I_k(i, j), \quad 1 < k \leq K, \quad (2.3)$$

where  $I_0(i, j) = I(i, j)$ . Having the wavelet representation as a set of  $K + 1$  images,  $W_1, \dots, W_K, I_K$ , also called the à trous wavelet representation, the reconstruction can be easily performed as

$$I(i, j) = I_K(i, j) + \sum_{k=1}^K W_k(i, j). \quad (2.4)$$

For denoising and object detection, the property of the wavelets to be localized in both space and frequency plays a major role, as it allows separation of the components of an image according to their size. The large values of  $W_k(i, j)$  correspond to some structures and the smaller ones usually to noise. The denoising is based on the modification of the images  $W_k(i, j)$ , by hard-thresholding the coefficients, and using the modified images  $\tilde{W}_k(i, j) = T_d(W_k)$  in the inverse transformation (2.4). Here, the

thresholding operator  $T_d : \mathcal{I} \rightarrow \mathcal{I}^{th}$  is defined as

$$I^{th}(i, j) = \begin{cases} I(i, j), & \text{if } |I(i, j)| \geq d, \\ 0, & \text{otherwise.} \end{cases} \quad (2.5)$$

The hard threshold  $d$  depends on the standard deviation of the wavelet coefficients  $\sigma_k$  per resolution level, and is usually taken to be  $3\sigma_k$ . Alternatively, the wavelet coefficients may be soft-thresholded according to more advanced schemes [47, 153]. However, for astronomical and also for biological images, soft thresholding should be avoided, as it leads to photometry loss in regard to all objects [153].

In order to reduce the dependence of the threshold  $d$  on the absolute values of the object and background intensities, the thresholding is often based on Bayesian analysis of the coefficient distributions using Jeffrey's noninformative prior [47] (also called the amplitude-scale-invariant), which is a nonlinear shrinkage rule that outperforms other famous shrinkage rules, including VisuShrink and SureShrink [47], and is given by

$$\tilde{W}_k(i, j) = W_k^{-1}(i, j)(W_k^2(i, j) - 3\sigma_k^2)_+, \quad (2.6)$$

where  $(x)_+ = \max\{x, 0\}$ . The threshold is proportional to the standard deviation of wavelet coefficients at each resolution level and it adaptively selects significant coefficients only. The modified filtered images  $\tilde{W}_k(i, j)$  are used in (2.4) for the inverse transformation to obtain the denoised image  $\mathcal{J}$ .

## 2.3.2 Unsupervised Signal Enhancement

### 2.3.2.1 Wavelet Multiscale Product

As was mentioned in Section 2.3.1.2, in the à trous wavelet representation, contrary to the frequently used orthogonal wavelet transform [91], the wavelet coefficients are correlated across the resolution levels (scales). This property is exploited by the detection approach based on the multiscale product [108], which uses the same image decomposition as in Section 2.3.1.2 and creates the multiscale product image as

$$P_K(i, j) = \prod_{k=1}^K W_k(i, j). \quad (2.7)$$

This transformation constitutes Step 2 in the general detection framework (Section 2.2.2). For better performance, the original algorithm [108] also includes the noise reduction step (Step 1) using the technique described in Section 2.3.1.2: the wavelet coefficients are hard-thresholded per scale,  $\tilde{W}_k(i, j) = T_{d_k}(W_k(i, j))$ , with the threshold  $d_k = k_d \sigma_k$ ,  $k_d = 3$ , and the modified coefficients  $\tilde{W}_k(i, j)$  are used in (2.7).

This method uses the fact that the real objects are represented by a small number of wavelet coefficients that are correlated across the scales. Contrarily, the coefficients that are due to noise are randomly distributed and are not propagated across scales. As a result, the image  $P_K(i, j)$ , which is the grayscale classification map  $\mathcal{C}$ , is thresholded with  $l_d$  and binarized. The connected components in the binary map  $\mathcal{C}_B$  are considered as detected objects (Step 3). In the original algorithm [108],  $l_d = 1.0$  and

no thresholds on the cluster size  $v_d$  in the thresholded and binarized  $P_K(i, j)$  were imposed [108]. In summary, this method has three parameters,  $(l_d, k_d, K)$ , that are not directly related to the object appearance. Recently, a modification of the described method, which uses the Gaussian kernel at several scales instead of B3-splines, was proposed for segmentation and analysis of nuclear components in stem cells [176].

### 2.3.2.2 Top-Hat Filter

Another class of methods that are used for detection of bright spots in the presence of widely varying background intensities is known as top-hat filters [20, 21]. Such filters are dynamic thresholding operators, rather than the similarly named image transformation from mathematical morphology. The latter transformation selects extended objects with sufficiently narrow parts, rather than compact objects, as does the top-hat filter considered here.

The filter discriminates the spots by their round shape and predetermined information about their intensity and size. At each pixel location,  $(i, j)$ , the average image intensity  $\bar{I}_{\text{top}}$  and  $\bar{I}_{\text{brim}}$  are calculated for pixels within two circular regions  $D_{\text{top}}$  and  $D_{\text{brim}}$ , respectively, defined as

$$D_{\text{top}}^{i,j} = \{(i', j') : (i - i')^2 + (j - j')^2 < R_{\text{top}}^2\}, \quad (2.8)$$

$$D_{\text{brim}}^{i,j} = \{(i', j') : R_{\text{top}}^2 < (i - i')^2 + (j - j')^2 < R_{\text{brim}}^2\}, \quad (2.9)$$

where the radius  $R_{\text{top}}$  corresponds to the “top” of the “hat” and is set to the maximum expected spot radius. The brim radius,  $R_{\text{brim}}$  ( $R_{\text{brim}} > R_{\text{top}}$ ), is often taken to be the shortest expected distance to the neighboring spot. If the difference  $\bar{I}_{\text{top}} - \bar{I}_{\text{brim}}$  is larger than some threshold  $H_{th}$ , the original image intensity  $I(i, j)$  for that position  $(i, j)$  is copied to the classification map  $\mathcal{C}$ ,  $\mathcal{C}(i, j) = I(i, j)$ , otherwise  $\mathcal{C}(i, j) = 0$ . The procedure is repeated for each pixel, and the binary map  $\mathcal{C}_B$  (Step 3) is obtained as  $\mathcal{C}_B(i, j) = 1$  if  $\mathcal{C}(i, j) \neq 0$ , and  $\mathcal{C}_B(i, j) = 0$  otherwise. The connected components are counted without any size or shape threshold.

The height  $H_{th}$  of the top above the brim is set to the minimum intensity that a spot must rise above its immediate background. It can also be related to the minimum local SNR that we are willing to deal with. If the detection of all the objects with local SNR  $> a$  is required, because for lower SNRs the detector would produce a lot more false positives and contaminate the analysis, the threshold  $H_{th}$  can be fixed to  $a\sigma_{\text{brim}}$ , where  $\sigma_{\text{brim}}$  is the standard deviation of the intensity distribution in the region  $D_{\text{brim}}$ .

In summary, the described algorithm has only three parameters,  $(H_{th}, R_{\text{top}}, R_{\text{brim}})$ , which can be related to the object appearance. The noise reduction (Step 1) in this case is implicitly done while calculating the average image intensity  $\bar{I}_{\text{top}}$  and  $\bar{I}_{\text{brim}}$ . The averaging decreases the variance in the estimation of the noisy object and background intensity levels and improves the robustness and performance of the method. A slightly modified version of the filter, called the top-hat box filter [20], uses a square mask for the regions  $D_{\text{top}}$  and  $D_{\text{brim}}$  and is computationally faster, but in the present context this is not an important advantage.

### 2.3.2.3 Spot-Enhancing Filter

The optimal filter for enhancing subresolution particles and reducing correlated noise in microscopy images is the whitened matched filter, which is well approximated by the Laplacian of a Gaussian (LoG) [128]. In this case, the convolution kernel  $(2\sigma_L^2 - i^2 - j^2)\sigma_L^{-4}G_{\sigma_L}$  is used in (2.2) to obtain the image  $\mathcal{J}$ , where the filter parameter  $\sigma_L$  must be tuned to the size of the particles. The filter combines Steps 1 and 2 and operates as a local background subtraction technique that preserves object-like structures and removes the background and noise. The filter can be made computationally fast by separable implementation [128]. The result of LoG filtering, the image  $\mathcal{J}$ , is used as the classification map  $\mathcal{C}$ , which is thresholded with  $l_d$  to locate the objects. This detection procedure has two parameters,  $(\sigma_L, l_d)$ , and is similar to the top-hat filter (Section 2.3.2.2), with the difference that here the convolution kernel, also called the “Mexican hat”, represents a continuous version of the top-hat filter mask.

### 2.3.2.4 Grayscale Opening Top-Hat Filter

Similar to the method above (Section 2.3.2.2), this top-hat filter uses the opening operation from mathematical morphology [138, 147, 149]. In order to improve the detector performance, the original image  $\mathcal{I}$  is first smoothed with the Gaussian kernel with scale  $\sigma$  (Step 1) and the grayscale opening of  $\mathcal{J}$  with a structuring element  $A$  is done, producing the image  $\mathcal{J}_A$ , where in our case a flat disk of radius  $r_A$  is used. The radius  $r_A$  is related to the size of the largest objects that we would like to detect. The top-hats are obtained after the subtraction  $\mathcal{C} = \mathcal{J} - \mathcal{J}_A$  (which concludes Step 2), and the whole transformation acts as a background subtraction method that leaves only compact structures smaller than the disk  $A$ , or extended objects with sufficiently narrow parts, rather than compact objects only, as does the top-hat filter. The resulting image  $\mathcal{C}$  is thresholded at level  $l_d$  (Step 3), and then all the connected components are counted. Additional filtering with  $v_d$  can be done if the size of the connected components should be taken into account. Thus, this method has four parameters,  $(\sigma, r_A, l_d, v_d)$ , all of which can be related to the object appearance.

### 2.3.2.5 H-Dome Based Detection

Another approach borrowed from grayscale mathematical morphology is based on the  $h$ -dome transformation [177], which was used in our previous works on subresolution particle tracking to design a detection scheme for track initiation and termination [142, 146]. The transformation has the interesting property that all the detected objects end up having the same maximum intensity in the transformed image, which we exploited to build a fast probabilistic tracker that outperforms current deterministic methods [146] and at the same time has the same tracking accuracy as the computationally more expensive particle filtering approaches for tracking [141, 146].

For this method we assume that the intensity distribution in the image  $\mathcal{I}$  is formed by  $N_o$  objects (bright spots), modeled using (2.1), background structures (also called clutter) with intensity distribution  $B(i, j)$ , and possibly spatially correlated additive or multiplicative noise  $\eta(i, j)$ . The main problem is to accurately estimate the number of real objects  $N_o$  and the object positions  $(x_l, y_l)^T$ ,  $l = \{1, \dots, N_o\}$ , in the presence

of inhomogeneous background structures and noise. The algorithm also consists of three steps: filtering,  $h$ -dome transformation, and “sampling” (signal thresholding). First, the image  $\mathcal{I}$  is LoG filtered with scale  $\sigma_L$ , which enhances the signal in the places where objects are present and performs local background subtraction (Step 1). The scale  $\sigma_L$  can be related to the size of the objects to be detected, and in our experiments is equal to 2.5 pixels (125 nm). Then, grayscale reconstruction [177] is performed on the LoG-filtered image  $\mathcal{J}$  with mask image  $\mathcal{J} - h$ , where  $h > 0$  is a constant (Step 2). As a result, the original image is decomposed into the reconstructed image  $\mathcal{B}_\sigma$  and the so-called  $h$ -dome image  $\mathcal{H}_\sigma$ :

$$\mathcal{I}_\sigma(i, j) = \mathcal{H}_\sigma(i, j) + \mathcal{B}_\sigma(i, j). \quad (2.10)$$

Geometrically speaking, similar to local background subtraction, the  $h$ -dome transformation extracts bright structures by “cutting off” the intensities of height  $h$  from the top, around local intensity maxima, producing “dome”-like structures. Contrary to top-hat filtering [177], this does not involve any shape or size criteria. The image  $\mathcal{B}_\sigma$  represents the nonuniform background structures, and image  $\mathcal{H}_\sigma$  contains the objects and all the smaller noise structures.

After the transformation, the maximum intensity of those Gaussian-like objects is approximately  $h$ , and for the noise structures the amplitude is less than  $h$  [146]. This transformed image  $\mathcal{H}_\sigma$  is used as a probability map for the final step of the algorithm (Step 3): the sampling. During this step, all the pixel values in  $\mathcal{H}_\sigma$  are raised to the power  $s$  in order to compensate for the broadening of the original object intensity distributions by the convolution with the LoG filter, and to create a highly peaked function that resembles the probability density function (pdf) of the object location distribution. The parameter  $s$  can be related to the maximum and minimum object size and the scale  $\sigma_L$  [146]. The function  $H_\sigma^s(i, j) = (\mathcal{J}(i, j) - \mathcal{B}_\sigma(i, j))^s$  is used in our framework as a so-called importance sampling function [9], denoted by  $q(i, j|\mathcal{I})$ , that describes which areas of the image most likely contain the objects. We sample  $N$  position-samples from  $q(i, j|\mathcal{I})$  using systematic resampling [9],  $\mathbf{x}^l \sim q(i, j|\mathcal{I})$ , where  $l = \{1, \dots, N\}$  and  $\mathbf{x} = (i, j)$ , in order to estimate the object positions using Monte Carlo methods. Then, the mean-shift algorithm [34] is used to cluster the samples  $\mathbf{x}^l$ , resulting in  $M$  clusters. For each cluster, the mean position  $\mathbf{x}_c = (i_c, j_c)$  and the variance  $R_c$  are computed using only the  $N_c$  samples  $\mathbf{x}^l$  belonging to that cluster:

$$\begin{aligned} \mathbf{x}_c &= \mathbb{E}[\mathbf{x}_c^l] = N_c^{-1} \sum_{l=1}^{N_c} \mathbf{x}_c^l, \\ R_c &= \mathbb{E}[(\mathbf{x}_c^l - \mathbf{x}_c)(\mathbf{x}_c^l - \mathbf{x}_c)^T]. \end{aligned} \quad (2.11)$$

The following two criteria are used to distinguish between real objects and other structures: 1) the number of samples  $N_c$  in the cluster should be larger than the number of samples in case of sampling from the uniform intensity distribution in the image region occupied by the cluster, and 2) the determinant of the covariance matrix of the cluster,  $\det R_c$ , must be less than  $\sigma_M^4/s^2$ , where  $\sigma_M$  characterizes the maximum object size that we are interested in. These two thresholds are motivated by the fact that the elements of the estimated covariance matrix  $R_c$  using the samples generated from the intensity distribution of the real objects, are bounded from above by  $(\sigma_{\max}^2 +$

$\sigma_L^2)s^{-1}$ . The samples that came from noise have approximately the same variance ( $R_c \approx \sigma_L^2 \mathbf{I}s^{-1}$ ), where  $\mathbf{I}$  is the identity matrix, but since the intensity amplitude  $\ll h$ , the number of samples  $N_c$  in the corresponding cluster will be below the mentioned threshold. The clutter on the other hand, having possibly high intensity values ( $\approx h$ ), produces a large number of samples, but the variance in those clusters is higher than in the case of the largest real object characterized by  $\sigma_M$ .

The parameters  $\sigma_L$  and  $\sigma_M$  of this detection method can be related to the object appearance. The height  $h$  is related to the SNR in the same way as in the case of the top-hat filter (Section 2.3.2.2). The method is fairly insensitive to the free parameters  $s$  and  $N$  [142, 146] (above some minimum, sensible values, which can be found experimentally and then fixed, these parameters primarily affect the computational cost of the method, not its accuracy). Thus, in summary, this method depends mainly on three parameters,  $(\sigma_L, \sigma_M, h)$ , that need to be tuned to the application.

### 2.3.2.6 Image Features Based Detection

The last unsupervised method that we consider in this study is based on using some additional image information during Step 2 that would help to distinguish the spots from the clutter. As was shown previously [141, 161], the incorporation of local curvature information can be used to build a reasonably good detector for image data with  $\text{SNR} > 4$ . The true spots in the image are characterized by a combination of convex intensity distributions and a relatively high intensity. Noise-induced local maxima typically exhibit a random distribution of intensity changes in all directions, leading to a low local curvature [161]. These two discriminative features (intensity and curvature) are used in combination during Step 2 to create the grayscale classification map  $\mathcal{C}$  using the denoised image (Step 1)  $J(i, j) = (G_\sigma * I)(i, j)$  as follows:

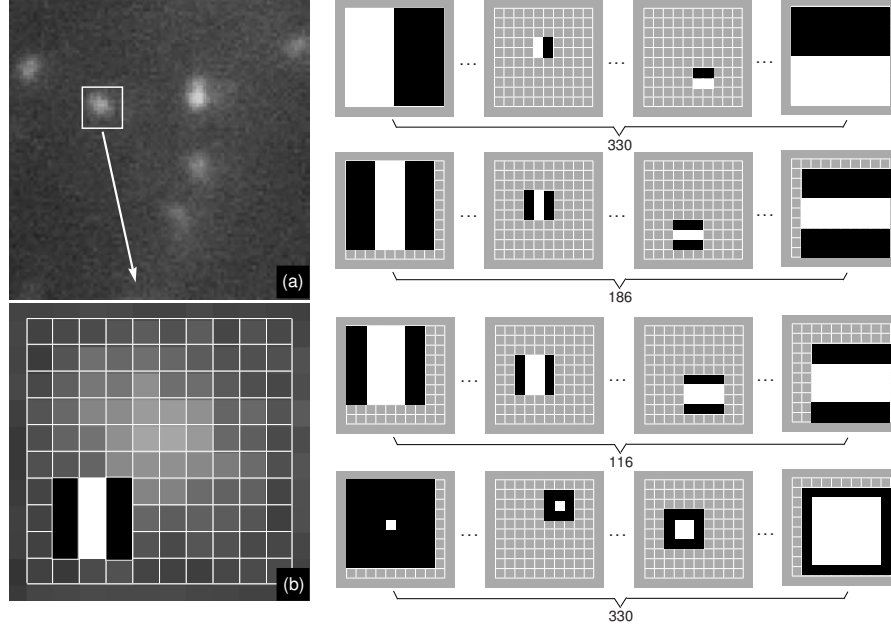
$$C(i, j) = J(i, j)\kappa(i, j), \quad (2.12)$$

where the curvature  $\kappa(i, j)$  at each pixel of  $\mathcal{J}$  is given by the determinant of the Hessian matrix  $\mathbf{H}(i, j)$  [159], which itself is known to be a good blob detector [86]. The classification map  $\mathcal{C}$  again is binarized (Step 3) using the threshold  $l_d$  and possibly the size threshold  $v_d$  which are not directly related to the object appearance.

### 2.3.3 Supervised Signal Enhancement

In order to make our comparison study of spot detection methods more complete, we also included two machine learning (ML) techniques. The first one is the AdaBoost algorithm [178], which is frequently used for object detection in computer vision [50, 85, 178], and was recently shown to perform well also for spot detection in molecular bioimaging [73]. The second method is Fisher discriminant analysis (FDA), which is a classical and well-known linear classifier, but which has not been employed for spot detection in fluorescence microscopy up to now. It uses the same information as AdaBoost but is computationally less expensive and much easier to understand conceptually.





**Figure 2.3.** Examples of the Haar-like features that were used in the experiments to detect spots, and the numbers of all possible scaled and translated versions in  $10 \times 10$ -pixel subwindows of the image.

### 2.3.3.1 AdaBoost

This ML detection algorithm operates on small patches of the image around the hypothesized spot positions (Fig. 2.3(a)) and classifies the patches (Fig. 2.3(b)) as positive (object is present) or negative (object is absent) based on the combined response of several simple feature-based classifiers. Usually the feature-based systems are favored over pixel-based ones because they are much faster and can encode some domain knowledge. A set of  $N_F$  simple Haar-like features is used [111], which is overcomplete in comparison with the real Haar basis [91], and in our case consists of four kinds (four different rows in Fig. 2.3). For each feature  $\eta_l$ ,  $l = \{1, \dots, N_F\}$ , the feature value  $\xi(\eta_l)$  is a weighted difference between the sum of the pixels within two (black and white) rectangular regions. The weights are chosen in such a way that the value of the feature computed for constant-intensity images is zero. The number of possible features, which are scaled and translated versions of the features of each kind (Fig. 2.3), depends on the image patch size, and for  $10 \times 10$ -pixel image subwindows [73] is 962 (the number of features per kind is indicated below each feature row in Fig. 2.3). Using the integral images [178], the computation of the sums of pixels in the rectangular regions can be performed very fast.

Having the pool of  $N_F$  features  $\eta_l$ , and a training set consisting of  $N_T$  image patches labeled as positive and  $N_T$  patches labeled as negative, we selected a variant of the AdaBoost learning algorithm that can be used both to select a small subset of features and to train the classifier [178]. Such a choice was made on the basis



of recently published results of applying the AdaBoost algorithm in bioimaging [73]. The AdaBoost algorithm is used to boost the performance of a simple (weak) learning algorithm. The weak classifier is designed to select the single feature that best separates the positive and negative samples. In our case, this separation is accomplished by finding the appropriate threshold  $d_l$  for each feature  $\eta_l$  at every round during the training stage. With each run of the algorithm, one feature is selected and added to the set of best discriminating features. The number of runs, denoted by  $N_{AB}$ , is user-defined. It is known that the training error of the strong classifier approaches zero exponentially in the number of rounds [50].

The final strong classifier is a weighted linear combination of all selected weak classifiers. The classification map  $\mathcal{C}_B$  (Step 3) is constructed as follows. First, for each pixel  $(i, j)$  of  $\mathcal{I}$  the value of the feature  $\eta_{l'}$  is computed using the corresponding  $10 \times 10$ -pixel image subwindow centered at  $(i, j)$  and assigned to  $\mathcal{C}^{l'}(i, j)$ , where  $l'$  specifies one of the  $N_{AB}$  features that were selected during the training. This way, the image  $\mathcal{C}^{l'}$  is obtained. Then, the values in  $\mathcal{C}^{l'}$  are thresholded using the feature threshold  $d_{l'}$ , producing a binary version  $\mathcal{C}_B^{l'}$  of  $\mathcal{C}^{l'}$ . The procedure is repeated for all  $N_{AB}$  features, and the images  $\mathcal{C}_B^{l'}$ ,  $l' = 1, \dots, N_{AB}$ , are combined (with weights also learned during the training) into  $\mathcal{C}$ , which is then thresholded with the threshold  $l_d = 0.5$  [178], producing the map  $\mathcal{C}_B$ . In the final classification map, some additional thresholding using the size information  $v_d$  (not related to the notion of spot size) might be needed in order to remove small regions with misclassified pixels.

By applying the trained classifier to the image  $\mathcal{I}$  (Step 2), prefiltering (Step 1) is performed implicitly: the values of the features are the difference in average pixel values in the black and white rectangular regions. This averaging reduces the variance of the feature value estimation in a similar way as in the case of the top-hat filter (Section 2.3.2.2).

### 2.3.3.2 Fisher Discriminant Analysis

Discriminant analysis is a statistical technique which classifies objects into one of two or more groups based on a set of features that describe the objects [93]. We use FDA to classify the image patches in the same way as in the AdaBoost method (Section 2.3.3.1). For an image patch of size  $n \times n$  pixels, the  $n$  horizontal rows of pixels are concatenated into a 1-D (column) feature vector  $\mathbf{y}$  of size  $n^2$ . Having a labeled training dataset with positive and negative samples (image patches), the corresponding sets of features  $\{\mathbf{y}_1^l\}_{l=1}^{N_T}$  and  $\{\mathbf{y}_0^l\}_{l=1}^{N_T}$  are used to compute the mean  $\boldsymbol{\mu}_c$  and the covariance matrix  $\boldsymbol{\Sigma}_c$  for each class  $c = \{0, 1\}$ . The task of FDA is to find the linear transformation  $\mathbf{w}$  that maximizes the ratio

$$Q(\mathbf{w}) = \frac{(\mathbf{w}^T(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0))^2}{\mathbf{w}^T(\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_0)\mathbf{w}}. \quad (2.13)$$

In some sense,  $Q(\mathbf{w})$  is a measure of the SNR for the class labeling, where the numerator represents the between-class variation and the denominator represents the within-class variation. It can be shown that the optimal separation occurs when  $\mathbf{w} = (\boldsymbol{\Sigma}_1 + \boldsymbol{\Sigma}_0)^{-1}(\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)$  [93]. This concludes the training stage. During the classification stage, when FDA is applied to patches extracted from the image  $\mathcal{I}$  using

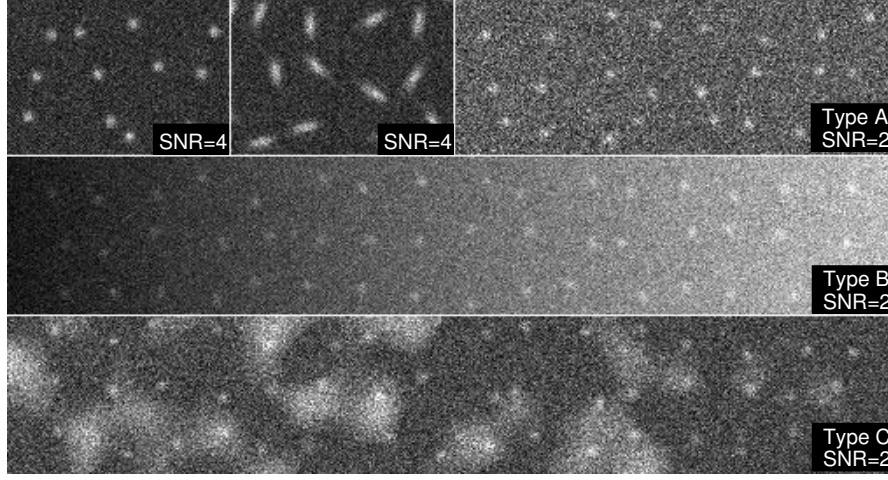
a sliding subwindow of size  $n \times n$  pixels, the patch is classified as positive (object is present,  $C_B(i, j) = 1$ ) if the condition  $|\mathbf{w}^T \mathbf{y} - \boldsymbol{\mu}_1| < |\mathbf{w}^T \mathbf{y} - \boldsymbol{\mu}_0|$  is satisfied, and as negative (object is absent,  $C_B(i, j) = 0$ ) otherwise.

The FDA classification procedure has an appealing interpretation as linear filtering (similar to (2.2)) with a kernel that is learned from the training data. The  $n^2$ -dimensional vector  $\mathbf{w}$  can be reshaped into an  $n \times n$  patch, similar to the image patch from which the feature vector  $\mathbf{y}$  is formed (see examples in Section 2.4.2, Fig. 2.17). In this case, the projection  $\mathbf{w}^T \mathbf{y}$ , which is performed using the sliding subwindow for each image pixel, is a convolution as in (2.2). The classification map  $\mathcal{C}$  is obtained by thresholding the convolution result at  $l_d = \frac{1}{2} \mathbf{w}^t (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_0)$ , which is obtained automatically because the training was performed beforehand.

### 2.3.4 Signal Thresholding and Performance Measures

As mentioned before, in order to locate and count the detected objects, the classification map  $\mathcal{C}$  is binarized using the threshold  $l_d$  (whose meaning depends on the method), and the connected components are searched for. Having the binary image  $\mathcal{C}_B$ , where  $C_B(i, j) = 1$  if  $C(i, j) > l_d$ , and  $C_B(i, j) = 0$  otherwise, we run the sequential scan labeling algorithm [66] in order to label the connected components and obtain the set of labels  $L(i, j)$  for all pixels, where  $L(i, j) \in \{0, \dots, M\}$ , with  $L = 0$  corresponding to the background and  $L \neq 0$  denoting one of the  $M$  detected objects. The center of mass,  $\mathbf{x}_m$ , is calculated for each of  $M$  objects, taking into account the pixels  $(i, j)$  and the image intensity  $I(i, j)$  for all  $(i, j)$  for which  $L(i, j) = m$ . The position is compared to the “ground truth”  $\mathbf{x}_m^0$  (known exactly in the case of synthetic images, and obtained manually by approximation in the case of real biological images). If  $\|\mathbf{x}_m^0 - \mathbf{x}_m\| < \Delta^0$ , the object is counted as a true positive (TP), otherwise the detected object is a false positive (FP). The number of false negatives (FN) is defined as  $N^0 - N_{\text{TP}}$ , where  $N^0$  is the number of objects in the ground truth and  $N_{\text{TP}}$  is the number of TPs. True negative (TN) is defined as accurate detection of the spot not to be an object. The number of TNs can be defined only for the ML approaches during the training stage. During the actual detection with any of the described methods, the number of TNs in the image data is undefined.

In order to measure the performance of the algorithms, we consider two common measures: the true-positive ratio (TPR),  $\text{TPR} = N_{\text{TP}} / (N_{\text{TP}} + N_{\text{FN}}) = N_{\text{TP}} / N^0$ , also called sensitivity, and the false-positive ratio (FPR),  $\text{FPR} = N_{\text{FP}} / (N_{\text{FP}} + N_{\text{TN}})$ . Because TN is not known for some methods, the modified version of FPR is used, given by  $\text{FPR}^* = N_{\text{FP}} / N^0$ . In this case, the standard receiver operating characteristic (ROC) curve cannot be built, and the modified version, called the free-response receiver operating characteristic (FROC) curve, is used [29, 30]. To demonstrate the sensitivity of TPR and  $\text{FPR}^*$  to parameters, for example the threshold  $l_d$ , we measure the values  $S_T = -(\partial \text{TPR} / \partial l_d)$  and  $S_F = -(\partial \text{FPR}^* / \partial l_d)$  at  $l_d = l_d^*$ . The threshold  $l_d^*$  is hereafter called “optimal” and corresponds to the value for which the  $\text{FPR}^* = 0.01$  (only 1% false positives). The value of TPR for  $l_d = l_d^*$  is denoted as  $\text{TPR}^*$ . Having  $S_T$  and  $S_F$ , we can compute the value  $\Delta \text{TPR} = 0.01 S_T l^*$ , which corresponds to the changes in TPR (around  $\text{TPR}^*$ ) when the parameter value  $l_d$  (or  $v_d$ ) is changed by 1% around  $l^*$  (or  $v^*$ ). Similarly,  $\Delta \text{FPR} = 0.01 S_F l^*$  can be introduced for the FPR.



**Figure 2.4.** Examples of synthetic images used in the experiments. The symmetrical Gaussian intensity profiles are embedded into uniform (Type A), gradient (Type B), and non-uniform (Type C) backgrounds.

## 2.4 Experimental results

The performance of the eight detection methods (six unsupervised and two supervised methods) described in the previous section was quantitatively evaluated using both synthetic images (Section 2.4.1) and real image data (Section 2.4.2) acquired for different biological studies. In the experiments, we studied the dependence of the performance (TPR and FPR\*) on parameter settings, type of object (perfectly round or slightly elongated), and image quality (SNR). Here we describe the experimental setups and the results.

### 2.4.1 Evaluation on Synthetic Image Data

#### 2.4.1.1 Simulation Setup

The described detection methods were evaluated using synthetic but realistic 2D images (of size  $512 \times 512$  pixels, with  $\Delta_x = \Delta_y = 50$  nm) containing intensity profiles of round and elongated objects modeled using (2.1) with  $\sigma_{\max} = \sigma_{\min} = 100$  nm for round objects, and  $\sigma_{\max} = 250$  nm,  $\sigma_{\min} = 100$  nm for elongated objects, for different levels of Poisson noise in the range of  $\text{SNR} = 2\text{--}4$ . Such SNRs are typical for the real image data acquired in our biological applications and are lower than the critical level of  $\text{SNR} = 4\text{--}5$ , at which several classical detection methods break down [26,32]. Here, SNR is defined as the difference in intensity between the object and the background, divided by the standard deviation of the object noise [32].

In order to estimate the performance of the algorithms, three types of images were created (see Fig. 2.4), for each type of object shape and for each SNR. In every image, 256 Gaussian intensity profiles were placed at positions  $\mathbf{x}_{i',j'}^0 = (16 + 30i' +$

$\mathcal{U}_{[-10,10]}, 16+30j'+\mathcal{U}_{[-10,10]})^T$ , where  $i' = 0, \dots, 15$ ,  $j' = 0, \dots, 15$ , and  $\mathcal{U}_{[-a,a]}$  denotes the uniform random generator within the interval  $(-a, a)$ . This way, the objects were randomly placed, with no overlaps in the intensity distributions. Type A images were constructed by adding a background level of 10, similar to previous studies [32]. To form the final noisy image, a Poisson noise generator was applied independently to every pixel of the noise-free image. In the case of Type B images, the background level increased linearly in the horizontal direction (see Fig. 2.4), from a value of 10 at the left image border to 50 at the right border. Taking into account that the variance of Poisson noise is intensity dependent, we corrected the object intensities accordingly prior to application of the noise generator in order to keep the SNR constant over the whole image. Finally, type C images mimic the intensity distribution in the presence of large (compared to object size) background structures (clutter), which are sometimes present in the real image data and can be either larger subcellular structures or acquisition artifacts. In this case, the pixel values were sampled from the normal distribution  $I_0(i, j) \sim \mathcal{N}(0, 150)$ . Then, the image was convolved with the Gaussian kernel  $G_{10}$  and thresholded at zero-level. The final image  $\mathcal{I}$  was obtained by adding to  $T_0(G_{10} * I_0)$  a constant background level of 10 plus the (SNR-adapted) object intensity profiles, followed by application of Poisson noise. Examples of synthetic images of all three types are shown in Fig. 2.4. In every experiment, the performance of the detection techniques for each object type was evaluated by accumulating the numbers of TP and FN for 16 images (each containing 256 ground truth objects) and averaging the results over the 4096 objects. The distance between the ground truth location and the object position estimated by the detector,  $\Delta^0$ , which defines if the detected object is a TP or FP, was fixed to  $\Delta^0 = 200$  nm (4 pixels).

#### 2.4.1.2 Wavelet Multiscale Product

For the performance evaluation of the wavelet multiscale product detector (further abbreviated as WMP), the parameters of the method (see Section 2.3.2.1) were fixed to the values described in the original paper [108]:  $l_d = 1$ ,  $K = 3$ ,  $k_d = 3$ . The performance measures TPR and FPR\* for the image data with SNR = 2 are shown in Table 2.1. In order to evaluate the sensitivity of the method to parameter changes, we varied the number of scales  $K$  and the wavelet coefficient threshold  $k_d$  in our experiments and studied their influence on the behavior of TPR and FPR\*. In the experiments, the grayscale classification map  $\mathcal{C}$  produced by the method was thresholded at  $l_d$ , and after binarization all the connected components were labeled as detected objects. Because the method produced quite fractured clusters of pixels, we used the morphological opening operator with a square  $3 \times 3$  mask (a  $5 \times 5$  mask yielded very similar results) in order to fill in the holes.

The main results of the sensitivity analysis for this method are shown in Fig. 2.5. They show that a value of  $K = 3$  is a good compromise to maximize performance for all three different data types together (Fig. 2.5(a)-(c)). The results also show that the performance of this method drops quite rapidly when the SNR decreases from 4 to 2 (Fig. 2.5(d)), and also when the background complexity increases (Fig. 2.5(e)-(f)). Table 2.2 shows the “optimal” values of  $k_d$  for different types of data for  $l_d = 1$ ,  $K = 3$ , and SNR = 2.

**Table 2.1.** Performance of the WPM detector using the original algorithm parameters at SNR = 2.

Image Type	Round Objects		Elongated Objects	
	TPR	FPR*	TPR	FPR*
A	0.33	0.001	0.34	0.013
B	0.18	0.001	0.20	0.010
C	0.21	0.015	0.25	0.017

**Table 2.2.** Optimal parameters and performance of the WPM detector at SNR = 2 and number of scales  $K = 3$ .

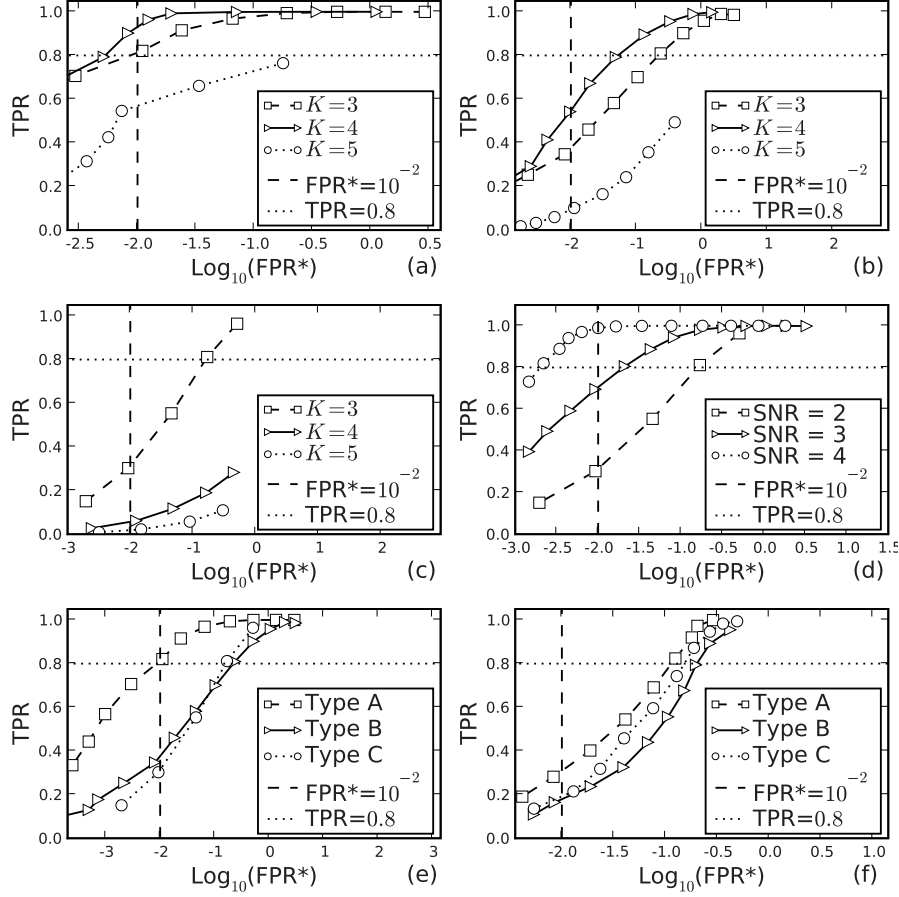
Image Type	Round Objects				Elongated Objects			
	$k_d^*$	TPR*	$S_T$	$S_F$	$k_d^*$	TPR*	$S_T$	$S_F$
A	2.22	0.81	.57	.04	3.06	.31	.61	.05
B	2.56	0.37	.56	.05	3.07	.17	.36	.05
C	2.89	0.30	.62	.09	3.17	.18	.39	.06

For comparison, we also applied the soft thresholding of the wavelet coefficients according to (2.6) instead of the original hard thresholding with  $k_d = 3$ . For round objects in Type C images at SNR = 2, using the hard threshold  $k_d = 3$ , we had  $FPR^* = 0.015$  and  $TPR = 0.21$ . The value of  $l_d$  was increased to 34 when the soft threshold (2.6) was used in order to obtain the same  $FPR^*$ , and the TPR in this case was equal to 0.25. For elongated objects the corresponding values were  $FPR^* = 0.017$  and  $TPR = 0.25$  for the hard thresholding, and  $TPR = 0.27$  for the soft thresholding.

Another experiment was conducted in order to investigate if the low performance of the WPM for SNRs around 2–3 was dependent on the type of noise (Poisson versus Gaussian). The variance-stabilizing Anscombe transform [7] was applied, which transforms the image intensities according to  $I(i, j) \rightarrow 2\sqrt{I(i, j)} + 3/8$ , and creates approximately Gaussian data of unit variance, provided that the mean value of the Poissonian data is more than 10 [7]. The experiments with the variance-stabilized (Gaussian) images showed no significant difference in TPR and FPR for all types of image data compared to the original (Poissonian) synthetic images.

#### 2.4.1.3 Top-Hat Filter

The performance of the top-hat filter (further abbreviated as TH) was evaluated using the same images as for the WPM detector. The brim radius,  $R_{\text{brim}}$ , which controls the local background estimation around the spot position, was fixed to 10 (see Section 2.3.2.2 for the parameters description). Varying this parameter in the range 8–12 did not influence the final results significantly, indicating that the local background estimation is quite robust. The TPR and  $FPR^*$  of the method for different  $R_{\text{top}}$  values, depending on  $H_{th}$ , are shown in Fig. 2.6. Again, holes within clusters (objects) in the binarized classification map  $\mathcal{C}_B$  were filled using the closing operation with a  $5 \times 5$  mask. All found clusters were considered as objects, regardless of cluster



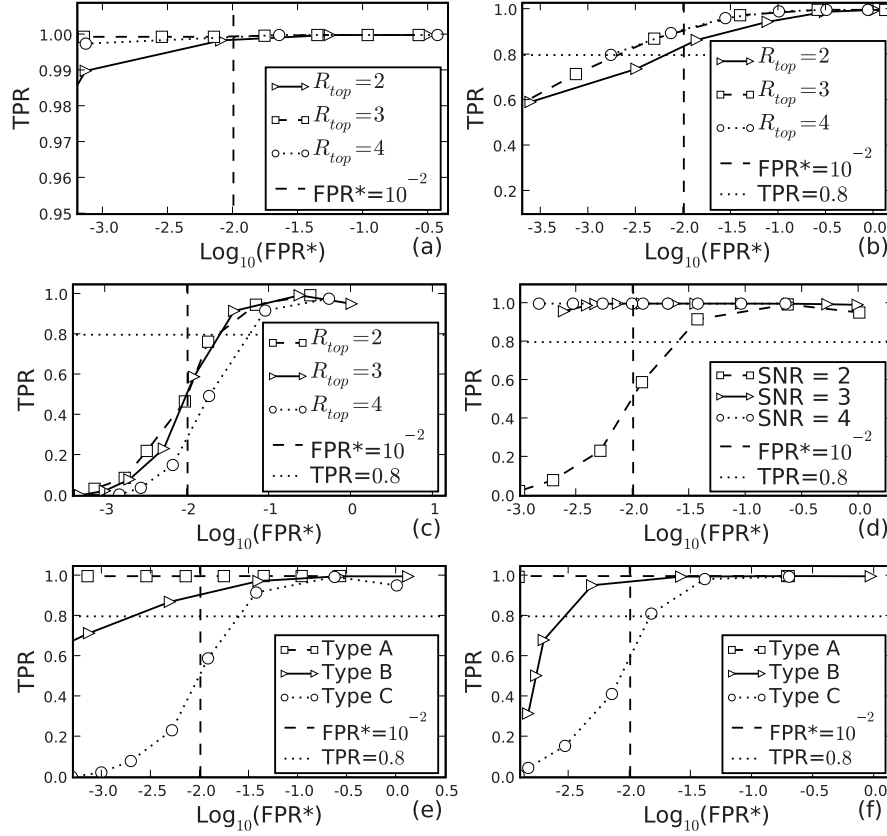
**Figure 2.5.** FROC curves for the WMP detector in the case of the round objects, depending on the wavelet coefficient threshold  $k_d$ , for Type A (a), Type B (b), and Type C (c) image data and different numbers of scales  $K$ , and the FROC curves for Type C data for different SNRs (d). The same type of FROC curves in the case of the round (e) and elongated (f) objects for different types of data, with SNR = 2 and  $K = 3$ .

size. The optimal values of  $H_{th}$  for all image types with SNR = 2 are shown in Table 2.3. The value of  $R_{top} = 3$  was chosen, which maximizes the TPR when  $FPR^* = 0.01$  for Type C data with both round and elongated objects.

#### 2.4.1.4 Spot-Enhancing Filter

The performance of the spot-enhancing filter (further abbreviated as SEF) using the synthetic images was studied depending on the values of the signal threshold  $l_d$  (see Section 2.3.2.3). The filter acts as a smoothing and local background subtraction technique at the same time (Steps 2 and 3). The only parameter is the scale of



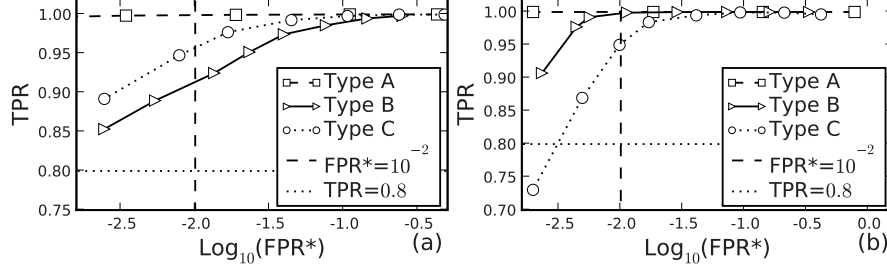


**Figure 2.6.** FROC curves for the TH detector in the case of the round objects, depending on the values of  $H_{th}$ , for several values of  $R_{top}$ , for Type A (a), Type B (b), and Type C (c) image data, and the FROC curves for Type C data for several SNRs (d). The same type of FROC curves in the case of the round (e) and elongated (f) objects depending on the values of  $H_{th}$  for different types of data, with  $SNR = 2$ ,  $R_{brim} = 10$ , and  $R_{top} = 3$ .

**Table 2.3.** Optimal parameters and performance of the TH detector at  $SNR = 2$  with radii  $R_{brim} = 10$  and  $R_{top} = 3$ .

Image Type	Round Objects				Elongated Objects			
	$H_{th}^*$	$TPR^*$	$S_T$	$S_F$	$H_{th}^*$	$TPR^*$	$S_T$	$S_F$
A	2.74	.99	.00	.05	2.95	.99	.00	.20
B	5.85	.88	.11	.03	5.75	.96	.04	.02
C	5.28	.48	.35	.01	5.62	.56	.38	.01

the convolution kernel,  $\sigma_L$ , which was tuned in order to get the highest TPR at  $FPR^* = 0.01$  in the case of Type C data. In the case of round objects, for  $\sigma_L$  values



**Figure 2.7.** FROC curves for the SEF detector in the case of round (a) and elongated (b) objects, depending on the values of the threshold  $H_{th}$  and the type of image data, at SNR = 2 and optimal scales  $\sigma_L = 2.5$  (for round objects) and  $\sigma_L = 3.1$  (for elongated objects).

**Table 2.4.** Optimal parameters and performance for the SEF detector at SNR = 2 and optimal scales  $\sigma_L = 2.5$  (for round objects) and  $\sigma_L = 3.1$  (for elongated objects).

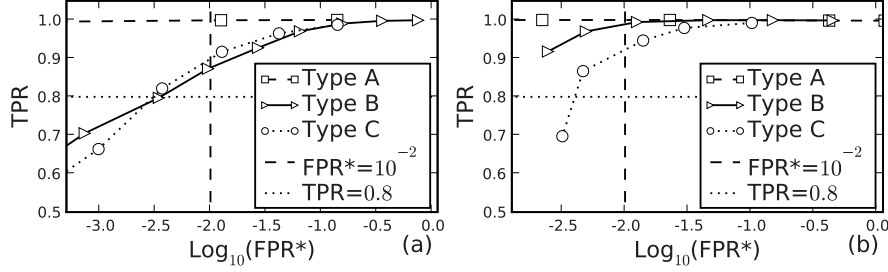
Image Type	Round Objects				Elongated Objects			
	$l_d^*$	TPR*	$S_T$	$S_F$	$l_d^*$	TPR*	$S_T$	$S_F$
A	0.85	.99	.01	.15	0.55	.99	.00	.16
B	1.84	.91	.35	.08	1.21	.99	.07	.06
C	1.22	.95	.29	.09	0.99	.95	.34	.07

$\{1.5, 2, 2.5, 3, 3.5\}$ , the corresponding TPR values were  $\{0.52, 0.9, 0.95, 0.9, 0.65\}$ , and thus  $\sigma_L = 2.5$  was used in the experiments. In the case of elongated objects, for  $\sigma_L$  in  $\{2.5, 3, 3.5, 4\}$ , the corresponding TPR values were  $\{0.75, 0.86, 0.92, 0.74\}$ , and  $\sigma_L = 3.1$  was used. All clusters in the binary classification map after signal thresholding were counted as objects, and the values  $l_d^*$  and corresponding TPR\*,  $S_T$ , and  $S_F$ , for which  $FPR^* = 0.01$ , are shown in Fig. 2.7 and Table 2.4. Again, the value  $l_d^*$  represents the optimal threshold, for which  $FPR^* = 0.01$ , with corresponding TPR denoted as TPR\*.

#### 2.4.1.5 Grayscale Opening Top-Hat Filter

This detection method from grayscale morphology (further abbreviated as MTH) is a robust local background subtraction technique. Its performance was not influenced significantly by changes of the mask size,  $r_A$ , in the range (3, 5) (see the parameter description in Section 2.3.2.4). The input images were first smoothed with the Gaussian kernel at  $\sigma = 2$ . The radius of the mask was fixed to  $r_A = 5$ , which means that all image structures of size smaller than the size of the disk  $A$  would be translated to the detection map  $\mathcal{C}$ . Two thresholds, one on the intensity amplitude and one on the object size, could be applied for the object extraction from  $\mathcal{C}$ . The latter threshold is crucial if the clutter consists of possibly elongated narrow structures, which would be considered as objects by this detector (see Section 2.3.2.4). We studied the dependence of TPR and  $FPR^*$  only on the intensity threshold  $l_d$ , as in the synthetic images





**Figure 2.8.** FROC curves for the MTH detector in the case of round (a) and elongated (b) objects, depending on the values of intensity threshold  $l_d$  for different types of image data, at  $\text{SNR} = 2$ , and with mask radius  $r_A = 5$  and Gaussian prefiltering at  $\sigma = 100$  nm.

**Table 2.5.** Optimal parameters and performance for the MTH detector at  $\text{SNR} = 2$  and with mask radius  $r_A = 5$  and Gaussian prefiltering at  $\sigma = 100$  nm.

Image Type	Round Objects				Elongated Objects			
	$l_d^*$	$\text{TPR}^*$	$S_T$	$S_F$	$l_d^*$	$\text{TPR}^*$	$S_T$	$S_F$
A	2.1	.99	.00	.04	2.1	.99	.00	.04
B	3.5	.87	.18	.06	4.1	.98	.05	.02
C	2.2	.88	.31	.03	3.2	.91	.15	.02

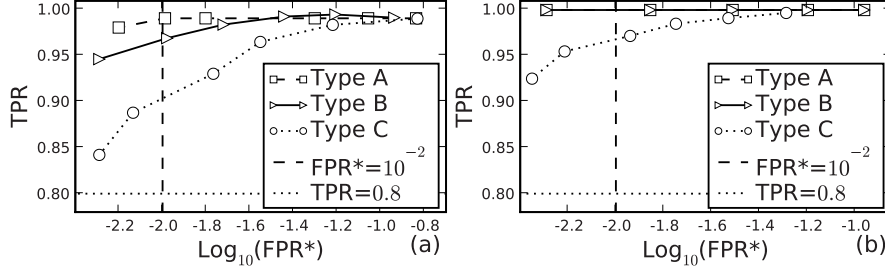
there are no clutter structures smaller than the object size. In this case, either intensity thresholding can be used without size thresholding, or a low intensity threshold can be used with further thresholding on the size. The values  $l_d^*$ , and corresponding  $\text{TPR}^*$ ,  $S_T$ , and  $S_F$ , for which  $\text{FPR}^* = 0.01$ , are shown in Fig. 2.8 and Table 2.5.

#### 2.4.1.6 H-Dome Based Detection

The method based on the  $h$ -dome transformation (further referred as HD) was evaluated depending on the dome height  $h$ . The parameters of the method (see Section 2.3.2.5) were fixed to  $\sigma_L = 2.5$ ,  $\sigma_M = 6$ ,  $s = 6$ , and  $N = 5000$ , which maximize the TPR for the Type C image data at  $\text{FPR}^* = 0.01$ . The results of the experiments are shown in Fig. 2.9. As described, the method estimates the object position and the variance of that estimation using a sampling procedure, bypassing the explicit creation of the map  $\mathcal{C}$  [146]. The values  $h^*$  and corresponding  $\text{TPR}^*$ ,  $S_T$ , and  $S_F$ , for which  $\text{FPR}^* = 0.01$ , are shown in Fig. 2.9 and Table 2.6.

#### 2.4.1.7 Image Features Based Detection

This scheme (further abbreviated as IFD) creates the classification map  $\mathcal{C}$  during Step 2 by combining the image intensities with local curvature information (see Section 2.3.2.6). Two types of the map  $\mathcal{C}$  were considered in the experiments (with the resulting methods abbreviated as  $\text{IFD}_1$  and  $\text{IFD}_2$  respectively). In the first case,  $\mathcal{C}$



**Figure 2.9.** FROC curves for the HD detector in the case of round (a) and elongated (b) objects, depending on the values of the dome height  $h$  for different types of image data, at SNR = 2, and with parameters  $\sigma_L = 2.5$ ,  $\sigma_M = 6$ ,  $s = 6$ , and  $N = 5000$ .

**Table 2.6.** Optimal parameters and performance for the HD detector at SNR = 2 for parameters  $\sigma_L = 2.5$ ,  $\sigma_M = 6$ ,  $s = 6$ , and  $N = 5000$ .

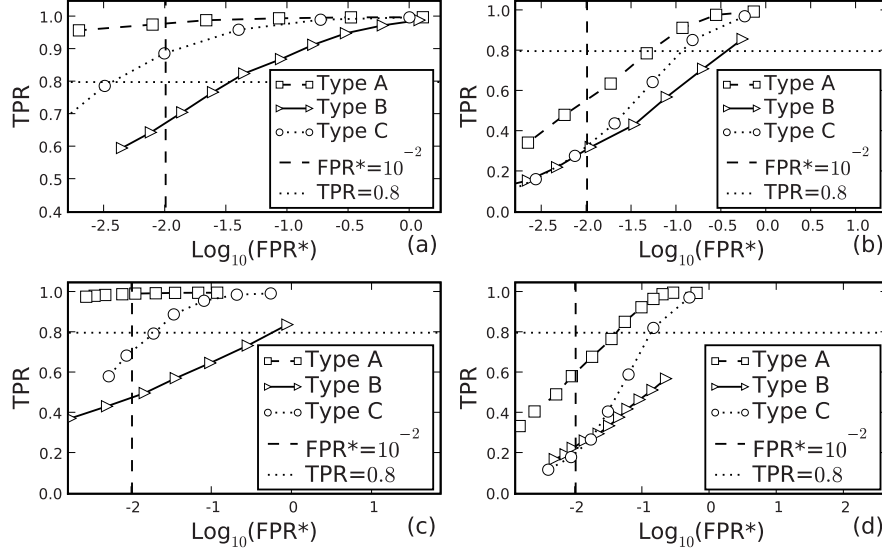
Image Type	Round Objects				Elongated Objects			
	$h^*$	TPR*	$S_T$	$S_F$	$h^*$	TPR*	$S_T$	$S_F$
A	1.6	.99	.11	.05	1.4	.99	.01	.09
B	1.6	.97	.22	.05	1.4	.99	.01	.09
C	1.6	.90	.21	.05	1.2	.97	.16	.05

is given by the determinant of the Hessian matrix,  $\det \mathbf{H}$ , calculated at each pixel, with smoothing scale  $\sigma$  [159]. The second type of classification map  $\mathcal{C}$  is obtained by pixel-wise multiplication of the values  $\det \mathbf{H}(i, j)$  with the intensity values  $J(i, j)$  (2.2). In the experiments, we used  $\sigma = 2$ , and the results are shown in Fig. 2.10 and Table 2.7.

#### 2.4.1.8 AdaBoost

In order to test the performance of the ML approaches, starting with AdaBoost (abbreviated as AB) for the detection of round objects, we constructed a pool of 962 Haar-like features (see Section 2.3.3.1) using a  $10 \times 10$  pixel subwindow, which was previously reported as optimal for similar applications [73]. Experiments with other subwindow sizes in the range of 8-12 pixels showed no significant difference in performance. For the detection of elongated objects, the subwindow size was fixed to  $13 \times 13$  pixels, which consequently gives 2366 features. Even though the characteristic size of the elongated objects is doubled (compared to the round objects), the use of larger subwindow sizes, for example  $21 \times 21$  pixels, degraded the AdaBoost performance. With the high spot density, the larger subwindows included the neighboring objects (equally frequently in the positive and negative training sets) and caused the problem with defining a clear decision boundary for these ML approach.

For the training stage, separate sets of synthetic images were created, and 4096 positive and 4096 negative samples ( $10 \times 10$  pixels) were extracted from each image type (A, B and C) containing round objects. The same training procedure was re-



**Figure 2.10.** FROC curves for the IFD<sub>1</sub> detector in the case of the round (a) and elongated (b) objects, depending on the values of the threshold  $l_d$  and the type of image data, at SNR = 2, and for smoothing scale  $\sigma = 2$ . The same curves for IFD<sub>2</sub> in the case of round (c) and elongated (d) objects.

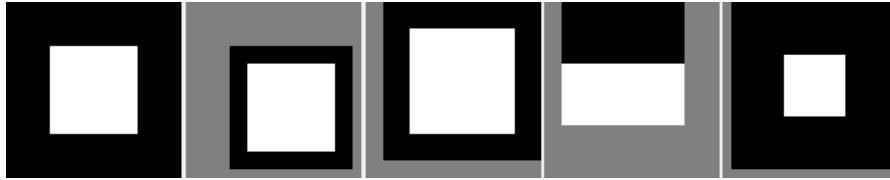
**Table 2.7.** Optimal parameters and performance for the IFD detectors at SNR = 2 and for smoothing scale  $\sigma = 2$ .

Image Type	Round Objects				Elongated Objects			
	$l_d^*$	TPR*	$S_T$	$S_F$	$l_d^*$	TPR*	$S_T$	$S_F$
IFD <sub>1</sub>								
A	.12	.98	0.67	.68	.21	.53	5.17	.42
B	.58	.67	1.23	.12	.71	.31	1.02	.06
C	.18	.89	2.51	.16	.28	.31	3.21	.26
IFD <sub>2</sub>								
A	1.33	.99	.03	.03	3.06	.59	.32	.03
B	33.34	.46	.01	.00	43.36	.23	.01	.00
C	1.95	.71	.36	.03	6.33	.19	.08	.01

peated for elongated objects. Four types of training were performed: using the samples from each image type separately, and using the combined training dataset, where 4095 samples were selected (in total) from type A, B and C images in equal proportions. The training was based on SNR = 2 (the worst case considered in this chapter). Training using higher-SNR images resulted in worse performance on lower-SNR images, as the number of features selected by AdaBoost became too small. Each trained classifier was applied separately to the synthetically created test images of all three types, with SNR in the range 2–4, and the classification results (sensitivity (TPR)

**Table 2.8.** Sensitivity and specificity of AdaBoost classification.

SNR	Image Type A		Image Type B		Image Type C	
	TPR	Spec.	TPR	Spec.	TPR	Spec.
Trained using type A data (SNR = 2)						
2	0.994	0.995	0.999	0.930	0.965	0.987
3	1.0	0.996	1.0	0.922	1.0	0.989
4	1.0	0.995	1.0	0.919	1.0	0.992
Trained using type B data (SNR = 2)						
2	0.914	1.0	0.991	0.977	0.690	1.0
3	1.0	0.999	1.0	0.977	0.998	0.999
4	1.0	0.999	1.0	0.977	1.0	0.999
Trained using type C data (SNR = 2)						
2	0.996	0.992	0.999	0.902	0.999	0.979
3	1.0	0.990	1.0	0.910	1.0	0.982
4	1.0	0.991	1.0	0.901	1.0	0.982
Trained using type A, B, C data combined (SNR = 2)						
2	0.988	0.998	0.998	0.942	0.962	0.994
3	1.0	0.997	1.0	0.939	1.0	0.995
4	1.0	0.998	1.0	0.940	1.0	0.993

**Figure 2.11.** Example of the top-five features that were selected by AdaBoost in the case of the Type A training data.

and specificity) for 4096 positive and 4096 negative patches, extracted from these test images, are given in Table 2.8. In the experiments, the number of AdaBoost runs,  $N_{AB}$ , which corresponds to the number of features selected and used by the classifier, was fixed to 5. The top-five features selected during the training are shown in Fig. 2.11.

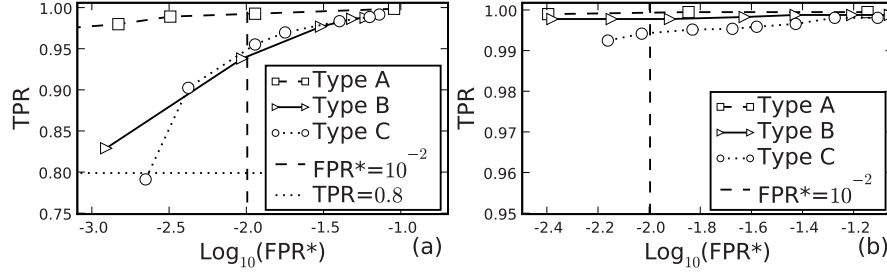
The behavior of the sensitivity and specificity was also investigated depending on the number of Haar-like features,  $N_{AB}$ , that are used for the classification. For this analysis, combined training (using the data of type A, B, and C) was performed, and the classifier was separately applied to the test data of each type. The results for different values of  $N_{AB}$  are shown in Table 2.9, where the last three rows also show the performance of the classifier trained using a reduced training set of 1002 combined samples (334 of each type).

In all these performance evaluation experiments, the classifier was applied to

**Table 2.9.** Sensitivity and specificity of AdaBoost classification depending on the number of runs.

SNR	Image Type A		Image Type B		Image Type C	
	TPR	Spec.	TPR	Spec.	TPR	Spec.
$N_{AB} = 5$						
2	0.988	0.998	0.998	0.942	0.962	0.994
3	1.0	0.997	1.0	0.939	1.0	0.995
4	1.0	0.998	1.0	0.940	1.0	0.993
$N_{AB} = 10$						
2	0.991	0.998	0.999	0.946	0.965	0.994
3	1.0	0.998	1.0	0.944	1.0	0.996
4	1.0	0.998	1.0	0.944	1.0	0.993
$N_{AB} = 20$						
2	0.991	0.999	0.999	0.953	0.965	0.994
3	1.0	0.998	1.0	0.957	1.0	0.996
4	1.0	0.998	1.0	0.954	1.0	0.996
$N_{AB} = 5$ and 1002 training samples						
2	0.991	0.999	0.999	0.953	0.965	0.994
3	1.0	0.998	1.0	0.957	1.0	0.996
4	1.0	0.998	1.0	0.954	1.0	0.996

image patches extracted from the positive and negative test images. In order to evaluate the performance of actual *detection* using this machine learning approach, we applied the classifier to each pixel in the images (based on a window of size  $10 \times 10$ -pixels around the pixel). The resulting classification map is a new image of the same size as the original, with each pixel being either “1” (if the corresponding image pixel was classified as belonging to an object) or “0” (if the pixel was classified as background). Before labeling the connected components and extracting the number of detected objects and their positions, the map was median-filtered with a round mask of radius 2 pixels in order to suppress too small clusters, and then a closing operation was applied with the  $3 \times 3$  structuring element to fill small holes. The FROC curves for this detection procedure depending on the size threshold  $v_d$  of the clusters in the binary classification map  $C_B$  in the case of round and elongated objects are shown in Fig. 2.12. The behavior of TPR and FPR\* depending on the number of features,  $N_{AB}$ , used in the detection is shown in Table 2.10. The parameters of the detection were optimized in order to get  $FPR^* = 0.01$  when  $N_{AB} = 50$ . After that, the number of features  $N_{AB}$  was reduced (see Table 2.10) and the behavior of the performance measures studied. The optimal parameter values for the size threshold  $v_d$  are shown in Table 2.11.



**Figure 2.12.** FROC curves for the AdaBoost detector in the case of the round (a) and elongated (b) objects, depending on the value of the size threshold  $v_d$ , at  $\text{SNR} = 2$ , and with  $N_{AB} = 50$ .

**Table 2.10.** Detection performance of AdaBoost depending on the number of selected features,  $N_{AB}$ , with training based on the combined image data (type A, B, and C) at  $\text{SNR} = 2$ .

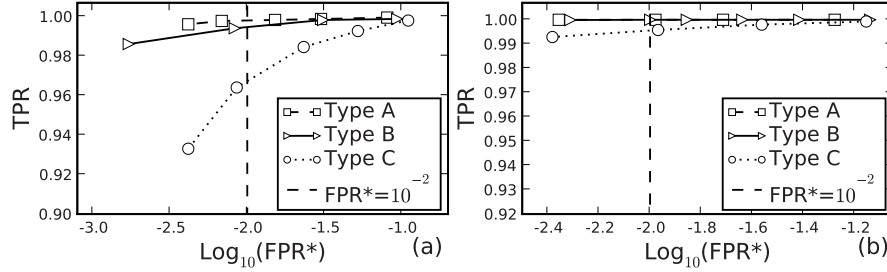
$N_{AB}$	Image Type A		Image Type B		Image Type C	
	TPR	FPR*	TPR	FPR*	TPR	FPR*
5	0.995	0.013	0.912	0.037	0.806	0.019
10	0.996	0.014	0.929	0.041	0.818	0.022
20	0.994	0.013	0.921	0.022	0.789	0.019
50	0.994	0.011	0.926	0.016	0.810	0.018

**Table 2.11.** Optimal size thresholding parameters and corresponding performance for AdaBoost at  $\text{SNR} = 2$ .

Image Type	Round Objects				Elongated Objects			
	$v_d^*$	TPR*	$S_T$	$S_F$	$v_d^*$	TPR*	$S_T$	$S_F$
A	3	.99	$10^{-3}$	$10^{-3}$	2	.99	$10^{-5}$	.10
B	31	.94	.01	$10^{-3}$	18	.99	$10^{-5}$	$10^{-3}$
C	30	.94	.01	$10^{-3}$	12	.99	$10^{-5}$	$10^{-3}$

#### 2.4.1.9 Fisher Discriminant Analysis

The classifier in this case (abbreviated as FDA) was trained using the same training data as in the case of AdaBoost. Using the labeled  $10 \times 10$  image patches (for the round objects) and  $13 \times 13$  patches (for the elongated objects), the kernels  $\mathbf{w}$  for both types of objects were obtained (see Fig. 2.17(d,e)). Then, the sliding subwindow was used in order to classify every pixel in the image  $\mathcal{I}$ . The method produces the binary classification map  $\mathcal{C}_B$  directly, so the performance of the detector was studied depending on the threshold  $v_d$  (which defines the size of the clusters of connected pixels in  $\mathcal{C}_B$ ), and not the signal threshold  $l_d$ . The results are shown in Fig. 2.13 and the optimal parameter values are presented in Table 2.12. The size threshold,



**Figure 2.13.** FROC curves for the FDA detector in the case of the round (a) and elongated (b) objects, depending on the values of the size threshold  $v_d$  and the type of image data, at SNR = 2.

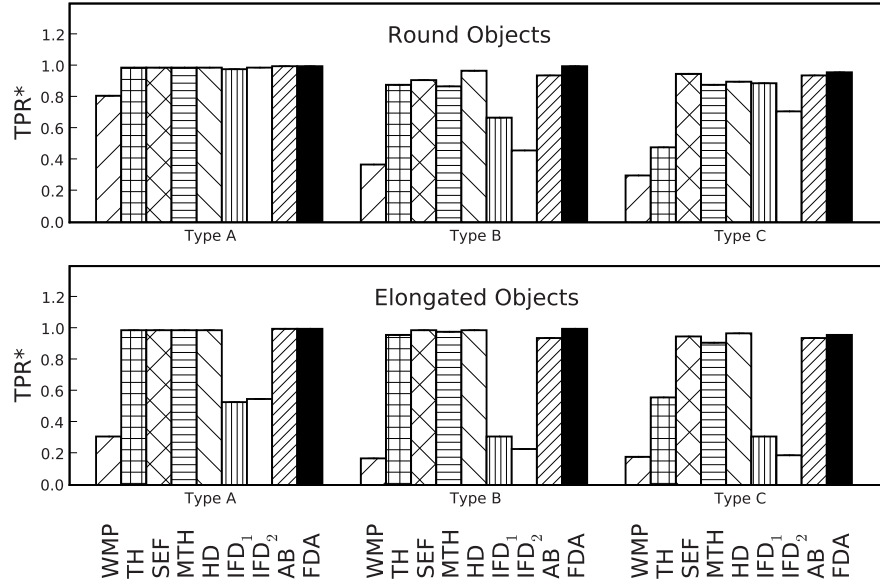
**Table 2.12.** Optimal size thresholding parameters and corresponding performance for the FDA detector at SNR = 2.

Image Type	Round Objects				Elongated Objects			
	$v_d^*$	TPR*	$S_T$	$S_F$	$v_d^*$	TPR*	$S_T$	$S_F$
A	4.6	.99	$10^{-5}$	.01	3.0	.99	$10^{-5}$	$10^{-2}$
B	8.8	.99	$10^{-3}$	.01	5.6	.99	$10^{-5}$	$10^{-2}$
C	9.8	.96	$10^{-2}$	.01	12.4	.99	$10^{-5}$	$10^{-3}$

which in principle is an integer number (the minimum number of pixels a cluster in  $\mathcal{C}_B$  should have to be considered an object), is real-valued in Table 2.12, due to the interpolation in order to obtain the value  $v_d^*$  for which  $\text{FPR}^* = 0.01$ .

#### 2.4.1.10 Comparison of All Detectors

The performance of all the described detectors was compared at the level of  $\text{FPR}^* = 0.01$  for the different image data at SNR = 2. The results are shown in Fig. 2.14. From the sensitivity analyses (see Tables 2.2-2.7, 2.11, 2.12), which was based on the comparison of  $\Delta\text{TPR}$  and  $\Delta\text{FPR}$  around the optimal signal thresholds for different detectors and data types revealed that the FDA and AB are superior to all other detectors and show the highest TPR\* and the lowest sensitivity for all image data (Type A, B and C, SNR = 2). The WMP demonstrated the worst performance and additionally showed high sensitivity to parameter changes, together with the TH detector, which demonstrated high performance only for Type A and B data. The IFDs are quite sensitive to parameter changes and do not have sufficiently high TPR in the case of the elongated objects. The HD, SEF and MTH demonstrate high TPR\* and low parameter sensitivity, but none of these three detectors is better than the other two for *all* types of data. Finally we observed that the difference in performance between the methods decreases when the SNR of the image data increases, and we found that for SNR > 5 all methods perform equally well (TPR = 1).



**Figure 2.14.** Maximum detection probabilities ( $TPR^*$ ) at the level  $FPR^* = 0.01$  for all the detectors applied to all three types of synthetic image data at  $SNR = 2$  in the case of the round (top) and elongated (bottom) objects.

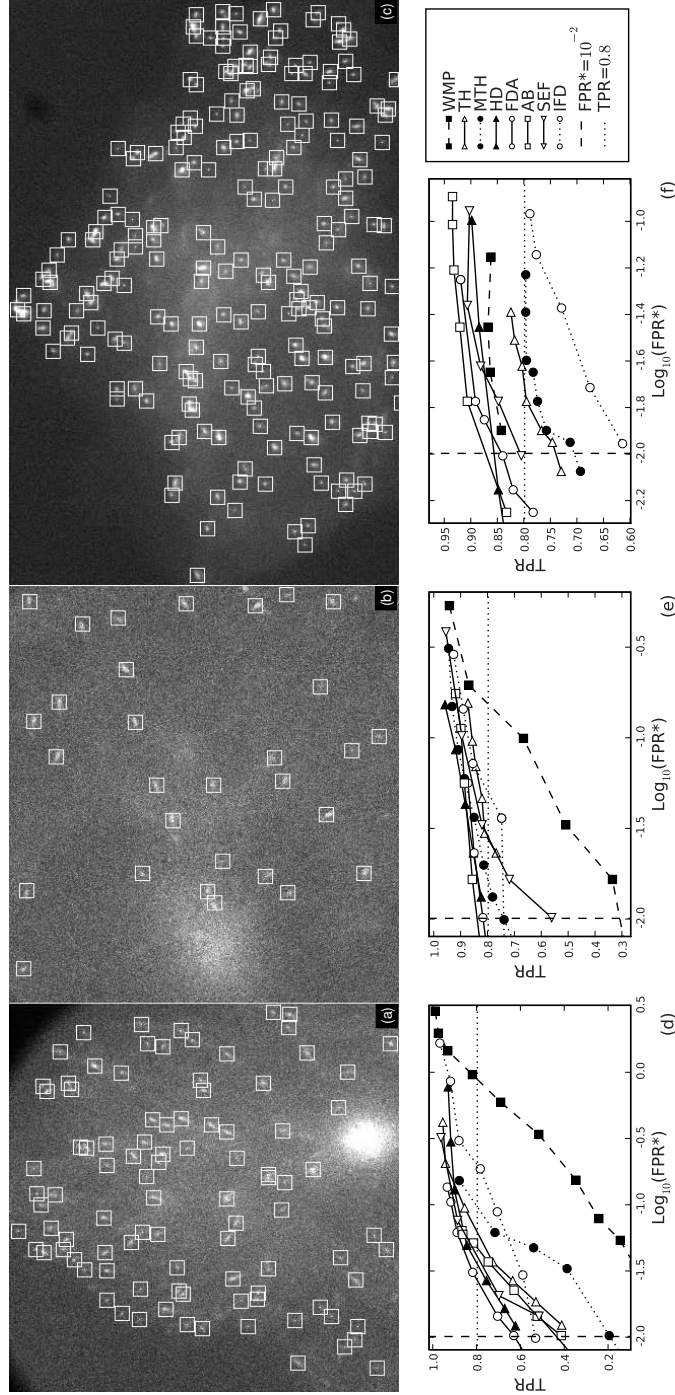
## 2.4.2 Evaluation on Real Image Data

### 2.4.2.1 Image Data

The described detection methods were also tested on real time-lapse fluorescence microscopy image data from several biological studies. The main goal of these studies was to estimate important kinematic parameters of subcellular particles in eukaryotic cells. To understand the molecular mechanisms underlying particle motility and distribution, it is essential to characterize in detail different dynamic properties, such as velocities, run lengths, and frequencies of pausing and switching of cytoskeletal tracks. This requires accurate tracking of individual particles, for which a wide variety of automatic tracking algorithms can be found in the recent literature [10, 17, 38, 52, 72, 74, 75, 128, 132, 141, 142]. In turn, these algorithms generally depend heavily on the performance of the spot detection stage, which forms an integral part of any tracking algorithm (see Section 2.1).

Two types of representative image data sets were selected for these experiments. The first showed moving microtubule (MT) plus-ends, which have a round or elongated appearance. MTs are hollow tubes (diameter of 25 nm) assembled from  $\alpha/\beta$ -tubulin heterodimers, which frequently switch between growth and shrinkage [80, 155]. The MT network is highly regulated and is essential to many cellular processes. In the experiments, growing ends of MTs were tagged with so-called plus-end-tracking proteins (+TIP), resulting in typical fluorescent “comet-like” dashes in the image se-





**Figure 2.15.** Examples of real fluorescence microscopy images (a, b, and c, compare Fig. 2.1) with manual spot annotation (white squares) by an expert biologist serving as ground truth. The corresponding FROCs (d, e, and f) of all the detection methods (with dependence on the same free parameters as in the experiments on the synthetic image data) are shown below the images (a, b, and c). In these plots, IFD represents the IFD<sub>1</sub> detector, which in the experiments on synthetic image data performed either similar to or better than the IFD<sub>2</sub> detector (see Fig. 2.14).

quences. In our study, COS-1 cells were cultured and transfected with GFP-tagged proteins [155]. A Zeiss 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  $512 \times 512$  pixels of size  $75 \times 75 \text{ nm}^2$  (see Fig. 2.15(a,b)).

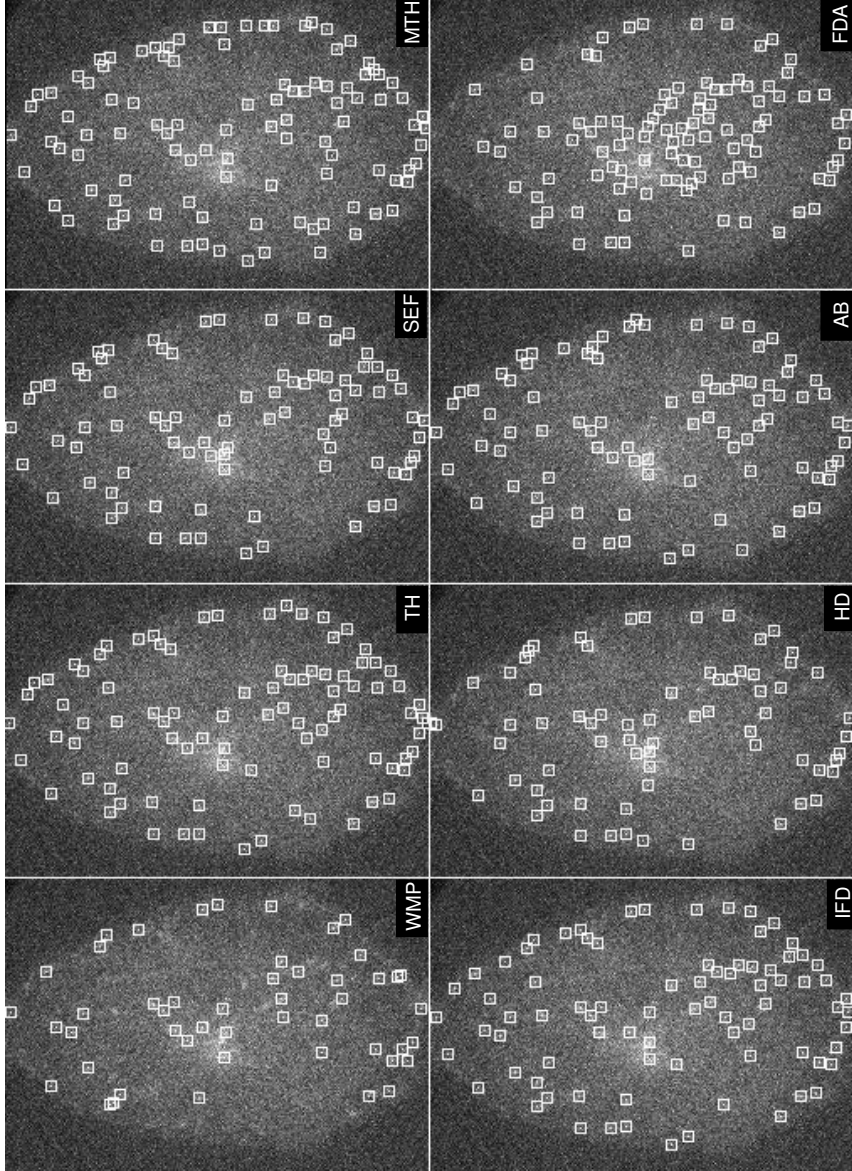
The second type of image data showed a variety of GFP-labeled vesicles (Rab6 and peroxisomes), which have a round shape in the images. In this case, HeLa cells and PEX3-GFP fusion were used [58]. The HeLa cell line is the oldest cell line and is widely used for many different studies. Many variants of the HeLa cell line exist, including HeLa-R, with a so-called “round” phenotype, and HeLa-L, with a “long” phenotype. HeLa-L cells were used to study the dynamic properties of vesicles, and HeLa-R cells to study microtubule dynamics, microtubule and cell cortex crosstalk, and exocytosis [58]. Images were acquired on a Zeiss Axiovert 200M inverted microscope at a rate of 0.83 frames per second. The image sequences consisted of 100 frames of  $1344 \times 1024$  pixels of size  $64 \times 64 \text{ nm}^2$  (see Fig. 2.15(c)).

#### 2.4.2.2 Experiments and Results

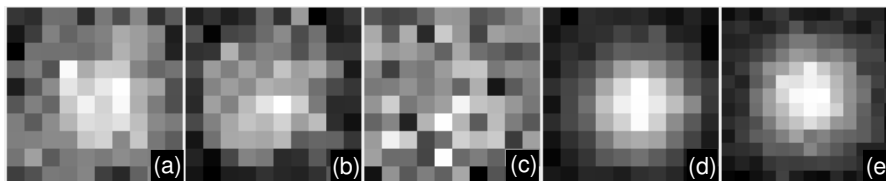
For the experiments on real image data, the parameters of each detection method (except the thresholds  $l_d$  and  $v_d$ ) were fixed to the same values as in the case of the experiments on synthetic data. Since the ground truth was not available for the real data, the results of the detection were analyzed by expert visual inspection and in comparison with manual analysis using MTrackJ [94].

The FROC plots for all the detection methods applied to two illustrative image data sets showing MTs (each image containing  $\approx 80$ – $100$  spots at  $\text{SNR} \approx 2$ – $4$ ) and one data set showing vesicles (containing  $\approx 250$  spots at  $\text{SNR} \approx 3$ – $8$ ) are shown in Fig. 2.15. For the latter data set, all detection methods performed reasonably well, including the WMP detector, which performed notably worse on the MT data. In all cases, the two ML detectors (FDA and AB) and the HD detector showed the best overall performance. For visual comparison, the kernels obtained by FDA for the three mentioned real image data sets, as well as for the two types of synthetic data sets are shown in Fig. 2.17, where, for example, Fig 2.17(c) depicts the fact that the vesicle appearance in our images (see Fig. 2.15(c)) is more diverse compared to the microtubule data (Fig. 2.15(a, b)).

As an example, the results of all methods applied to an MT data set with  $\text{SNR} \approx 2$  are shown in Fig. 2.16. Manual annotation was extremely laborious and tedious in this case: visual comparison of several neighboring time-frames in the image sequence was necessary in order to establish object presence. Based on visual inspection of the results, it was found that the HD detector yielded the largest number of TPs and the smallest number of FPs. Here, in order to test the robustness of the ML approaches, the training was done using positive and negative samples obtained from another dataset (see Fig. 2.1(b)) with  $\text{SNR} \approx 2$ – $3$ . The results of this experiment imply that FDA is more sensitive to the training data: if the training is done using image data with different imaging conditions (SNR), the performance of the classifier can degrade. The AdaBoost algorithm, on the other hand, is less sensitive.



**Figure 2.16.** Results of applying all the described detection methods to real fluorescence microscopy image data showing GFP+TIP-labeled MTs at  $\text{SNR} \approx 2$ . The HD detector yielded the largest number of TPs and the smallest number of FPs. Similar to Fig. 2.15, IFD represents the IFD<sub>1</sub> detector, which performed either similar to or better than the IFD<sub>2</sub> detector.



**Figure 2.17.** The FDA kernels for the MT data (a and b), vesicles (c), and the round and elongated objects from the synthetic data (d and e).

## 2.5 Discussion and Conclusions

In this chapter we have evaluated the performance of six unsupervised and two supervised detection methods that are frequently used in practice for the detection of small spots in fluorescence microscopy images. It was shown that all of the described methods follow a “three-step” signal processing procedure, but implement each of these steps in a specific way. In order to build an accurate and robust detector for a particular application, a careful selection of the algorithms for each of the steps is necessary. The results from experiments on synthetic images as well as real image data from two biological studies indicated that no detector outperforms all others in all considered situations. Overall, the supervised (machine learning) methods performed better on the synthetic images as well as on the real image data, but the differences in the performance were not large compared to some of the unsupervised methods.

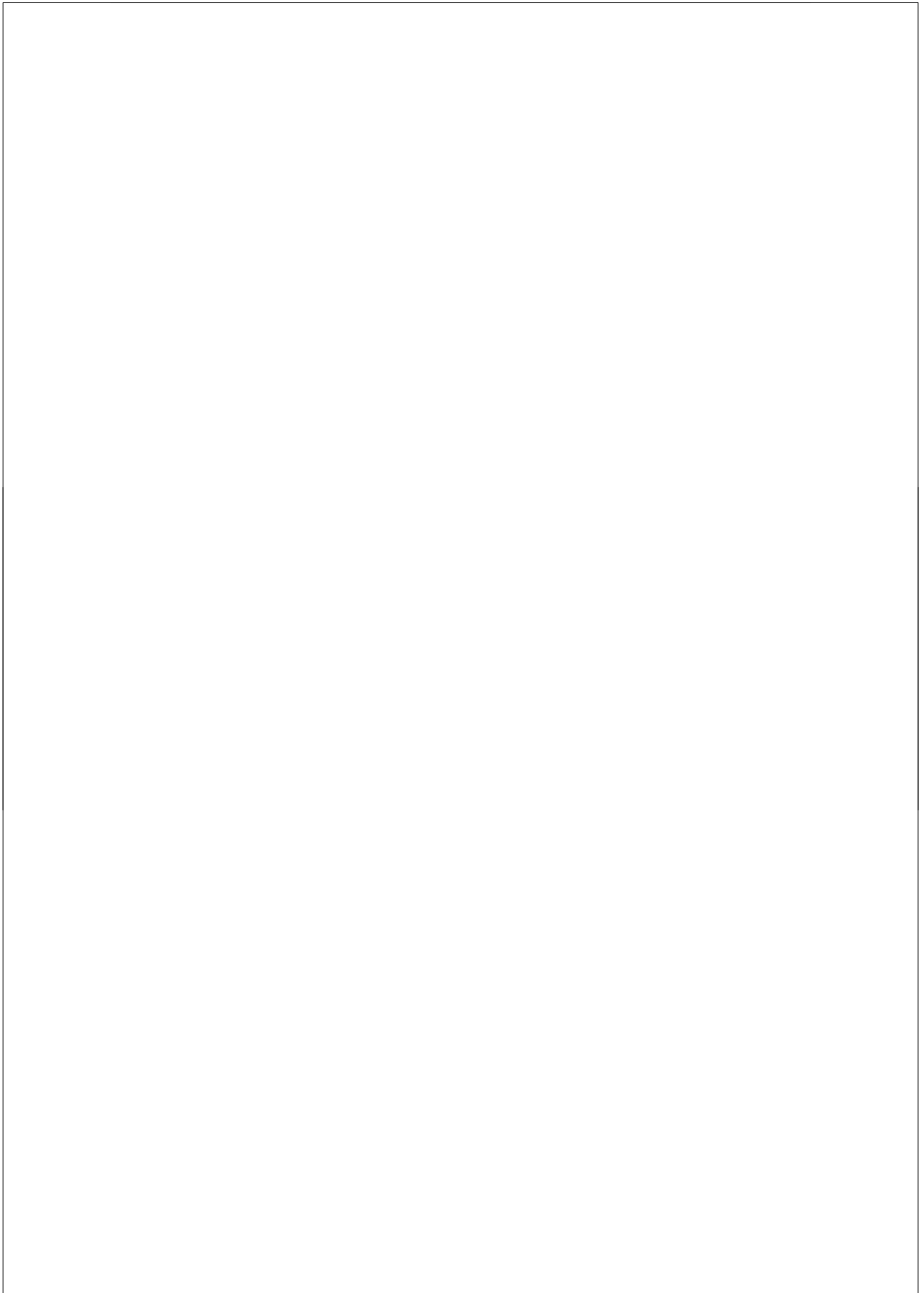
In order to study the influence of small changes in the parameter settings of the detection methods, a sensitivity analysis was carried out by computing the resulting rate of change in TPR (the true-positive ratio) and FPR (the false-positive ratio) around the empirically determined optimal signal threshold, for two types of objects (round and elongated). From the experiments on the synthetic images at very low SNR ( $\approx 2$ ), we found that the AB (AdaBoost) and the FDA (Fisher discriminant analysis) detectors are superior to all other detectors, in that they show the highest TPR (at very low FPR) and the lowest sensitivity to parameter changes, for all types of image data considered: uniform background (Type A), background gradient (Type B), and cluttered background structures (Type C). Of all the unsupervised detectors, the WMP (wavelet multiscale product) detector showed the worst overall performance and, additionally, high sensitivity to parameter changes. Similarly, the TH (top-hat based) detector showed high performance only for Type A and Type B data. The HD (*h*-dome), MTH (morphological top-hat), and SEF (spot-enhancing filter) based detectors showed high TPR and low parameter sensitivity, but none of them was better than the other two for all data types. Both variants of IFD (the image-feature based detector) were quite sensitive to parameter changes and did not show high TPR in the detection of elongated objects. Finally, we also observed from these experiments that for  $\text{SNR} > 3$ , the difference in performance of all the detectors rapidly decreases.

From the experiments on real fluorescence microscopy image data, it was confirmed that the actual performance of the detection methods depends on the application. For the microtubule data, which contained round or elongated objects of



almost identical sizes, we arrived at the same conclusions as in the case of the synthetic image data. For the vesicle data, however, the ranking of the detectors was found to be slightly different. These images have a higher SNR ( $\approx 3-8$ ) but contain spots of varying sizes. In this case, the detection methods that have parameters that explicitly relate to spot size, such as the TH and MTH detectors, showed quite poor performance. Once their parameters are set, these detectors expect spots to be of similar size. Similarly, the image-feature based IFD detector works well only when all the spots have very similar appearance in terms of the features considered. On the other hand, detectors such as SEF and HD do not model the spots exactly, and because of that allow some more variation in the appearance of spots. Moreover, the WMP detector, which also does not assume any specific object shape, demonstrated much better performance for such datasets.

Based on our extensive experiments, we conclude that when a detector with overall good performance is needed, the supervised AB or FDA detectors or the unsupervised HD detector are to be preferred. The main disadvantage of the supervised methods is that they require a training stage, which involves the extraction of positive and negative samples beforehand. As was shown, the training should not be done using only clearly visible spots in image regions with high local SNRs. On the contrary, in order to achieve good classification performance, it must also include a lot of hardly visible objects. Such manual annotation is extremely tedious, time consuming, and observer dependent. Spots may be more or less identical within one data set, but may differ in appearance from one data set to another, due to the different experimental and imaging conditions. Because of that, one would have to repeat the training (or correct it) when new data sets arrive. The preparation of training samples requires manual annotation of thousands of objects in order to achieve sufficient discriminating power, which itself is a manual detection that biologists would be happy to use, without considering further automated analysis. Taking this into account, the unsupervised HD detector is much easier to use in practice. Finally, when the SNR is sufficiently high ( $> 5$  as a rule of thumb), the other unsupervised detectors perform just as well, and require only minimal adjustment of their parameters to the specific application.



# Particle Filtering for Multiple Object Tracking in Dynamic Fluorescence Microscopy Images: Application to Microtubule Growth Analysis

---

*It is remarkable that a science which began with the consideration of games of chance should have become the most important object of human knowledge.*

— PIERRE-SIMON, MARQUIS DE LAPLACE  
*Théorie Analytique des Probabilités* (1812)

**Abstract** — Quantitative analysis of dynamic processes in living cells by means of fluorescence microscopy imaging requires tracking of hundreds of bright spots in noisy image sequences. Deterministic approaches, which use object detection prior to tracking, perform poorly in the case of noisy image data. We propose an improved, completely automatic tracker, built within a Bayesian probabilistic framework. It better exploits spatiotemporal information and prior knowledge than common approaches, yielding more robust tracking also in cases of photobleaching and object interaction. The tracking method was evaluated using simulated but realistic image sequences, for which ground truth was available. The results of these experiments show that the method is more accurate and robust than popular tracking methods. In addition, validation experiments were conducted with real fluorescence microscopy image data acquired for microtubule growth analysis. These demonstrate that the method yields results that are in good agreement with manual tracking performed by expert cell biologists. Our findings suggest that the method may replace laborious manual procedures.

---

Based upon: I. Smal, K. Draegestein, N. Galjart, W. Niessen, E. Meijering, “Particle Filtering for Multiple Object Tracking in Dynamic Fluorescence Microscopy Images: Application to Microtubule Growth Analysis”, *IEEE Transactions on Medical Imaging*, vol. 27, no. 6, pp. 789–804, 2008.

### 3.1 Introduction

In the past decade, advances in molecular cell biology have triggered the development of highly sophisticated live cell fluorescence microscopy systems capable of *in vivo* multidimensional imaging of subcellular dynamic processes. Analysis of time-lapse image data has redefined the understanding of many biological processes, which in the past had been studied using fixed material. Motion analysis of nanoscale objects such as proteins or vesicles, or subcellular structures such as microtubules (Fig. 3.1), commonly tagged with green fluorescent protein (GFP), requires tracking of large and time-varying numbers of spots in noisy image sequences [54, 95, 132, 135, 160, 161, 166]. Nowadays, high-throughput experiments generate vast amounts of dynamic image data, which cannot be analyzed manually with sufficient speed, accuracy and reproducibility. Consequently, many biologically relevant questions are either left unaddressed, or answered with great uncertainty. Hence, the development of automated tracking methods which replace tedious manual procedures and eliminate the bias and variability in human judgments, is of great importance.

Conventional approaches to tracking in molecular cell biology typically consist of two subsequent steps. In the first step, objects of interest are detected separately in each image frame and their positions are estimated based on, for instance, intensity thresholding [19], multiscale analysis using the wavelet transform [52], or model fitting [161]. The second step solves the correspondence problem between sets of estimated positions. This is usually done in a frame-by-frame fashion, based on nearest-neighbor or smooth-motion criteria [33, 171]. Such approaches are applicable to image data showing limited numbers of clearly distinguishable spots against relatively uniform backgrounds, but fail to yield reliable results in the case of poor imaging conditions [26, 32]. Tracking methods based on optic flow [13, 167] are not suitable because the underlying assumption of brightness preservation over time is not satisfied in fluorescence microscopy, due to photobleaching. Methods based on spatiotemporal segmentation by minimal cost path searching have also been proposed [17, 128]. Until present, however, these have been demonstrated to work well only for the tracking of a single object [128], or a very limited number of well-separated objects [17]. As has been observed [17], such methods fail when either the number of objects is larger than a few dozen, or when the object trajectories cross each other, which make them unsuitable for our applications.

As a consequence of the limited performance of existing approaches, tracking is still performed manually in many laboratories worldwide. It has been argued [95] that in order to reach similar superior performance as expert human observers in temporal data association, while at the same time achieving a higher level of sensitivity and accuracy, it is necessary to make better use of temporal information and (application specific) prior knowledge about the morphodynamics of the objects being studied. The human visual system integrates to a high degree spatial, temporal and prior information [23] to resolve ambiguous situations in estimating motion flows in image sequences. Here we explore the power of a Bayesian generalization of the standard Kalman filtering approach in emulating this process. It addresses the problem of estimating the hidden state of a dynamic system by constructing the posterior probability density function (pdf) of the state based on all available information, including

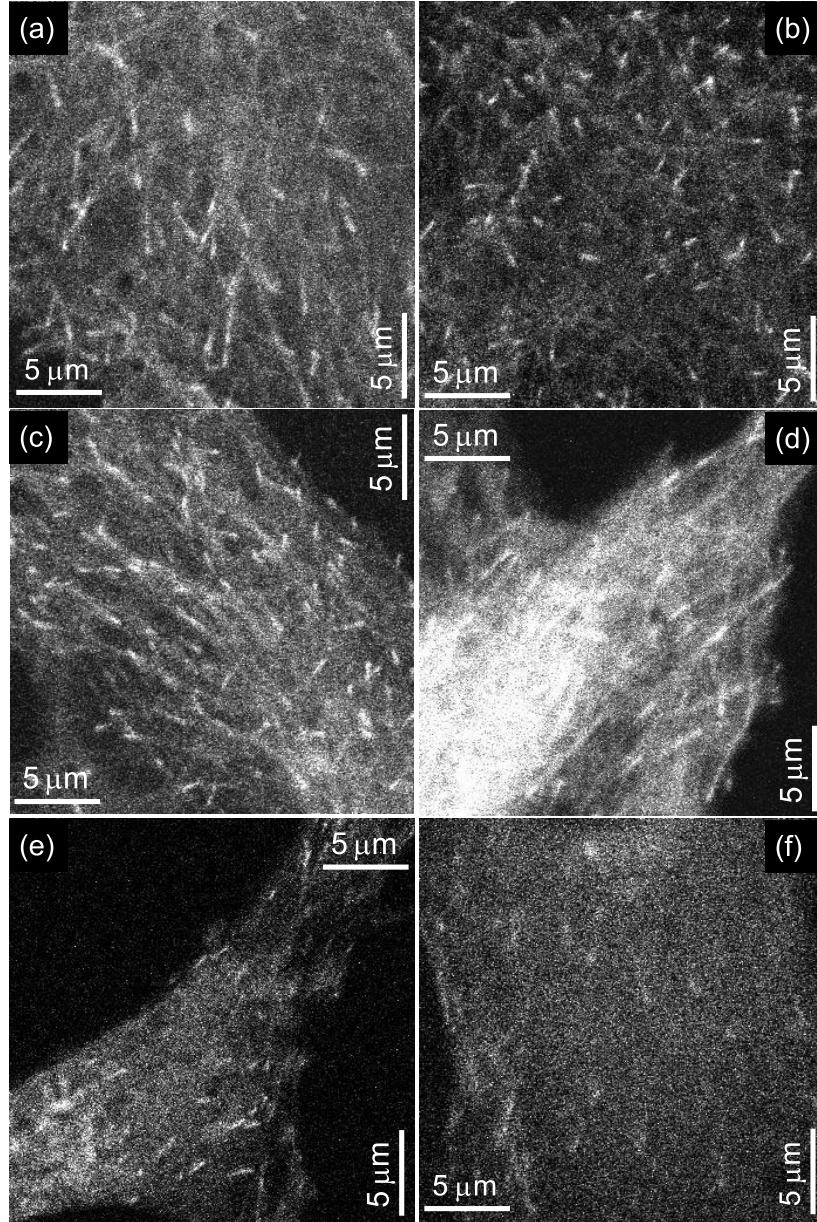


prior knowledge and the (noisy) measurements. Since this pdf embodies all available statistical information, it can be termed a complete solution to the estimation problem.

Bayesian filtering is a conceptual approach, which yields analytical solutions, in closed form, only in the case of linear systems and Gaussian statistics. In the case of non-linearity and non-Gaussian statistics, numerical solutions can be obtained by applying sequential Monte Carlo (SMC) methods [39], in particular particle filtering (PF) [9]. In the filtering process, tracking is performed by using a predefined model of the expected dynamics to predict the object states, and by using the (noisy) measurements (possibly from different types of sensors) to obtain the posterior probability of these states. In the case of multiple target tracking, the main task is to perform efficient measurement-to-target association, on the basis of thresholded measurements [15]. The classical data association methods in multiple target tracking can be divided into two main classes: unique-neighbor data association methods, as in the multiple hypothesis tracker (MHT), which associate each measurement with one of the previously established tracks, and all-neighbors data association methods, such as joint probabilistic data association (JPDA), which use all measurements for updating all track estimates [15]. The tracking performance of these methods is known to be limited by the linearity of the data models. By contrast, SMC methods that propagate the posterior pdf, or methods that propagate the first-order statistical moment (the probability hypothesis density) of the multitarget pdf [90], have been shown to be successful in solving the multiple target tracking and data association problems when the data models are nonlinear and non-Gaussian [68, 104].

Previous applications of PF-based motion estimation include radar- and sonar-based tracking [104, 175], mobile robot localization [39, 184], teleconferencing or video surveillance [115], and other human motion applications [31, 110, 186]. In most computer vision applications, tracking is limited to a few objects only [70, 89]. Most biological applications, on the other hand, require the tracking of large and time-varying numbers of objects. Recently, the use of PF in combination with level-sets [83] and active contours [139] has been reported for biological cell tracking. These methods outperform deterministic methods, but they are straightforward applications of the original algorithm [70] for single target tracking, and cannot be directly applied to the simultaneous tracking of many intracellular objects. A PF-like method for the tracking of proteins has also been suggested [183], but it still uses template matching for the linking stage, it requires manual initialization, and tracks only a single object. In this chapter, we extend our earlier conference reports [143, 144], and develop a fully automated PF-based method for robust and accurate tracking of multiple nanoscale objects in two-dimensional (2D) and three-dimensional (3D) dynamic fluorescence microscopy images. Its performance is demonstrated for a particular biological application of interest: microtubule growth analysis.

The chapter is organized as follows. In Section 3.2 we give more in-depth information on the biological application considered in this chapter, providing further biological motivation for our work. In Section 3.3 we present the general tracking framework and its extension to allow tracking of multiple objects. Next, in Section 3.4, we describe the necessary improvements and adaptations to tailor the framework to the application. These include a new dynamic model which allows dealing with object



**Figure 3.1.** Examples of microtubules tagged with GFP-labeled plus end tracking proteins (bright spots), imaged using fluorescence confocal microscopy. The images are single frames from six 2D time-lapse studies, conducted with different experimental and imaging conditions. The quality of such images typically ranges from  $\text{SNR} \approx 5\text{--}6$  (a-c) to the extremely low  $\text{SNR} \approx 2\text{--}3$  (d-f).

interaction and photobleaching effects. In addition, we improve the robustness and reproducibility of the algorithm by introducing a new importance function for data-dependent sampling (the choice of the importance density is one of the most critical issues in the design of a PF method). We also propose a new, completely automatic track initiation procedure. In Section 3.5, we present experimental results of applying our PF method to synthetic image sequences, for which ground truth was available, as well as to real fluorescence microscopy image data of microtubule growth. A concluding discussion of the main findings and their potential implications is given in Section 3.6.

## 3.2 Microtubule Growth Analysis

Microtubules (MTs) are polarized tubular filaments (diameter  $\approx 25$  nm) composed of  $\alpha/\beta$ -tubulin heterodimers. In most cell types, one end of a MT (the minus-end) is embedded in the so-called MT organizing center (MTOC), while the other end (the plus-end) is exposed to the cytoplasm. MT polymerization involves the addition of  $\alpha/\beta$ -tubulin subunits to the plus end. During MT disassembly, these subunits are lost. MTs frequently switch between growth and shrinkage, a feature called dynamic instability [37]. The conversion of growth to shrinkage is called catastrophe, while the switch from shrinkage to growth is called rescue. The dynamic behavior of MTs is described by MT growth and shrinkage rates, and catastrophe and rescue frequencies. MTs are fairly rigid structures having nearly constant velocity while growing or shrinking [48]. MT dynamics is highly regulated, as a properly organized MT network is essential for many cellular processes, including mitosis, cell polarity, transport of vesicles, and the migration and differentiation of cells. For example, when cells enter mitosis, the cdc2 kinase controls MT dynamics such that the steady-state length of MTs decreases considerably. This is important for spindle formation and positioning [173]. It has been shown that an increase in catastrophe frequency is largely responsible for this change in MT length [172].

Plus-end-tracking proteins, or +TIPs [137], specifically bind to MT plus-ends and have been linked to MT-target interactions and MT dynamics [4, 67, 80]. Plus-end-tracking was first described for overexpressed GFP-CLIP170 in cultured mammalian cells [114]. In time-lapse movies, typical fluorescent “comet-like” dashes were observed, which represented GFP-CLIP170 bound to the ends of growing MTs. As plus-end tracking is intimately associated with MT growth, fluorescently labeled +TIPs are now widely used to measure MT growth rates in living cells, and they are also the objects of interest considered in the present work. With fluorescent +TIPs, all growing MTs can be discerned. Alternatively, the advantage of using fluorescent tubulin is that all parameters of MT dynamics can be measured. However, in regions where the MT network is dense, the fluorescent MT network obscures MT ends, making it very difficult to examine MT dynamics. Hence, in many studies based on fluorescent tubulin [62, 127, 187], analysis is restricted to areas within the cells where the MT network is sparse. Ideally, one should use both methods to acquire all possible knowledge regarding MT dynamics, and this will be addressed in future work.

+TIPs are well positioned to perform their regulatory tasks. A network of inter-

acting proteins, including +TIPs, may govern the changes in MT dynamics that occur during the cell cycle [106]. Since +TIPs are so important and display such a fascinating behavior, the mechanisms by which +TIPs recognize MT ends have attracted much attention. In one view, +TIPs binds to newly synthesized MT ends with high affinity and detach seconds later from the MT lattice, either in a regulated manner or stochastically [114]. However, other mechanisms have also been proposed [4, 27, 67]. Measuring the distribution and displacement of a fluorescent +TIP in time may shed light on the mechanism of MT end binding. However, this is a labor intensive procedure if fluorescent tracks have to be delineated by hand, and very likely leads to user bias and loss of important information. By developing a reliable tracking algorithm we obtain information on the behavior of all growing MTs within a cell, which reveals the spatiotemporal distribution and regulation of growing MTs. Importantly, this information can be linked to the spatiotemporal fluorescent distribution of +TIPs. This is extremely important, since the localization of +TIPs reports on the dynamic state of MTs and the cell.

### 3.3 Tracking Framework

Before describing the details of our tracking approach, we first recap the basic principles of nonlinear Bayesian tracking in general (Section 3.3.1), and PF in particular (Section 3.3.2), as well as the extension that has been proposed in the literature to allow tracking of multiple objects within this framework (Section 3.3.3).

#### 3.3.1 Nonlinear Bayesian Tracking

The Bayesian tracking approach deals with the problem of inferring knowledge about the unobserved state of a dynamic system, which changes over time, using a sequence of noisy measurements. In a state-space approach to dynamic state estimation, the state vector  $\mathbf{x}_t$  of a system contains all relevant information required to describe the system under investigation. Bayesian estimation in this case is used to recursively estimate a time evolving posterior distribution (or filtering distribution)  $p(\mathbf{x}_t|\mathbf{z}_{1:t})$ , which describes the object state  $\mathbf{x}_t$  given all observations  $\mathbf{z}_{1:t}$  up to time  $t$ .

The exact solution to this problem can be constructed by specifying the Markovian probabilistic model of the state evolution,  $D(\mathbf{x}_t|\mathbf{x}_{t-1})$ , and the likelihood  $L(\mathbf{z}_t|\mathbf{x}_t)$ , which relates the noisy measurements to any state. The required probability density function  $p(\mathbf{x}_t|\mathbf{z}_{1:t})$  may be obtained, recursively, in two stages: prediction and update. It is assumed that the initial pdf,  $p(\mathbf{x}_0|\mathbf{z}_0) \equiv p(\mathbf{x}_0)$ , also known as the prior, is available ( $\mathbf{z}_{1:0} = \mathbf{z}_0$  being the set of no measurements).

The prediction stage involves using the system model and pdf  $p(\mathbf{x}_{t-1}|\mathbf{z}_{1:t-1})$  to obtain the prior pdf of the state at time  $t$  via the Chapman-Kolmogorov equation:

$$p(\mathbf{x}_t|\mathbf{z}_{1:t-1}) = \int D(\mathbf{x}_t|\mathbf{x}_{t-1})p(\mathbf{x}_{t-1}|\mathbf{z}_{1:t-1})d\mathbf{x}_{t-1}. \quad (3.1)$$

In the update stage, when a measurement  $\mathbf{z}_t$  becomes available, Bayes' rule is used to modify the prior density and obtain the required posterior density of the current



state:

$$p(\mathbf{x}_t | \mathbf{z}_{1:t}) \propto L(\mathbf{z}_t | \mathbf{x}_t) p(\mathbf{x}_t | \mathbf{z}_{1:t-1}). \quad (3.2)$$

This recursive estimation of the filtering distribution can be processed sequentially rather than as a batch, so that it is not necessary to store the complete data set nor to reprocess existing data if a new measurement becomes available [9]. The filtering distribution embodies all available statistical information and an optimal estimate of the state can theoretically be found with respect to any sensible criterion.

### 3.3.2 Particle Filtering Methods

The optimal Bayesian solution, defined by the recurrence relations (3.1) and (3.2), is analytically tractable in a restrictive set of cases, including the Kalman filter, which provides an optimal solution in case of linear dynamic systems with Gaussian noise, and grid based filters [9]. For most practical models of interest, SMC methods (also known as bootstrap filtering, particle filtering, and the condensation algorithm [70]) are used as an efficient numerical approximation. The basic idea here is to represent the required posterior density function  $p(\mathbf{x}_t | \mathbf{z}_{1:t})$  with a set of  $N_s$  random samples, or particles, and associated weights  $\{\mathbf{x}_t^{(i)}, w_t^{(i)}\}_{i=1}^{N_s}$ . Thus, the filtering distribution can be approximated as

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

where  $\delta(\cdot)$  is the Dirac delta function and the weights are normalized such that  $\sum_{i=1}^{N_s} w_t^{(i)} = 1$ . These samples and weights are then propagated through time to give an approximation of the filtering distribution at subsequent time steps.

The weights in this representation are chosen using a sequential version of importance sampling (SIS) [125]. It applies when auxiliary knowledge is available in the form of an importance function  $q(\mathbf{x}_t | \mathbf{x}_{t-1}, \mathbf{z}_t)$  describing which areas of the state-space contain most information about the posterior. The idea is then to sample the particles in those areas of the state-space where the importance function is large and to avoid as much as possible generating samples with low weights, since they provide a negligible contribution to the posterior. Thus, we would like to generate a set of new particles from an appropriately selected proposal function, i.e.,

$$\mathbf{x}_t^{(i)} \sim q(\mathbf{x}_t | \mathbf{x}_{t-1}^{(i)}, \mathbf{z}_t), \quad i = \{1, \dots, N_s\}. \quad (3.3)$$

A detailed formulation of  $q(\cdot | \cdot)$  is given in Section 3.4.6.

With the set of state particles obtained from (3.3), the importance weights  $w_t^{(i)}$  may be recursively updated as follows:

$$w_t^{(i)} \propto \frac{L(\mathbf{z}_t | \mathbf{x}_t^{(i)}) D(\mathbf{x}_t^{(i)} | \mathbf{x}_{t-1}^{(i)})}{q(\mathbf{x}_t^{(i)} | \mathbf{x}_{t-1}^{(i)}, \mathbf{z}_t)} w_{t-1}^{(i)}. \quad (3.4)$$

Generally, any importance function can be chosen, subject to some weak constraints [40, 126]. The only requirements are the possibility to easily draw samples from it

and evaluate the likelihood and dynamic models. For very large numbers of samples, this MC characterization becomes equivalent to the usual functional description of the posterior pdf.

By using this representation, statistical inferences, such as expectation, maximum a posteriori (MAP), and minimum mean square error (MMSE) estimators (the latter is used for the object position estimation in the approach proposed in this chapter), can easily be approximated. For example,

$$\hat{\mathbf{x}}_t^{\text{MMSE}} = \mathbb{E}_p[\mathbf{x}_t] = \int \mathbf{x}_t p(\mathbf{x}_t | \mathbf{z}_{1:t}) d\mathbf{x}_t \approx \sum_{i=1}^{N_s} \mathbf{x}_t^{(i)} w_t^{(i)}. \quad (3.5)$$

A common problem with the SIS particle filter is the degeneracy phenomenon, where after a few iterations, all but a few particles will have negligible weight. The variance of the importance weights can only increase (stochastically) over time [40]. The effect of the degeneracy can be reduced by a good choice of importance density and the use of resampling [9, 40, 125] to eliminate particles that have small weights and concentrate on particles with large weights (see [40] for more details on degeneracy and resampling procedures).

### 3.3.3 Multi-Modality and Mixture Tracking

It is straightforward to generalize the Bayesian formulation to the problem of multi-object tracking. However, due to the increase in dimensionality, this formulation gives an exponential explosion of computational demands. The primary goal in a multi-object tracking application is to determine the posterior distribution, which is multi-modal in this case, over the current *joint* configuration of the objects at the current time step, given all observations up to that time step. Multiple modes are caused either by ambiguity about the object state due to insufficient measurements, which is supposed to be resolved during tracking, or by measurements coming from multiple objects being tracked. Generally, MC methods are poor at consistently maintaining the multi-modality in the filtering distribution. In practice it frequently occurs that all the particles quickly migrate to one of the modes, subsequently discarding other modes.

To capture and maintain the multi-modal nature, which is inherent to many applications in which tracking of multiple objects is required, the filtering distribution is explicitly represented by an  $M$ -component mixture model [174]:

$$p(\mathbf{x}_t | \mathbf{z}_{1:t}) = \sum_{m=1}^M \pi_{m,t} p_m(\mathbf{x}_t | \mathbf{z}_{1:t}), \quad (3.6)$$

with  $\sum_{m=1}^M \pi_{m,t} = 1$  and a non-parametric model is assumed for the individual mixture components. In this case, the particle representation of the filtering distribution,  $\{\mathbf{x}_t^{(i)}, w_t^{(i)}\}_{i=1}^N$  with  $N = MN_s$  particles, is augmented with a set of component indicators,  $\{c_t^{(i)}\}_{i=1}^N$ , with  $c_t^{(i)} = m$  if particle  $i$  belongs to mixture component  $m$ . For the mixture component  $m$  we also use the equivalent notation  $\{\mathbf{x}_{m,t}^{(l)}, w_{m,t}^{(l)}\}_{l=1}^{N_s} =$

$\{\mathbf{x}_t^{(i)}, w_t^{(i)} : c_t^{(i)} = m\}_{i=1}^N$ . The representation (3.6) can be updated in the same fashion as the two-step approach for standard Bayesian sequential estimation [174].

### 3.4 Tailoring the Framework

Having presented the general framework for PF-based multiple object tracking, we now tailor it to our application: the study of MT dynamics. This requires making choices regarding the models involved as well as a number of computational and practical issues. Specifically, we propose a new dynamic model, which does not only cover spatiotemporal behavior but also allows dealing with photobleaching effects (Section 3.4.1) and object interaction (Section 3.4.2). In addition, we propose a new observation model and corresponding likelihood function (Section 3.4.3), tailored to objects that are elongated in their direction of motion. The robustness and computational efficiency of the algorithm are improved by using two-step hierarchical searching (Section 3.4.4), measurement gating (Section 3.4.5) and a new importance function for data-dependent sampling (Section 3.4.6). Finally, we propose practical procedures for particle reclustering (Section 3.4.7) and automatic track initiation (Section 3.4.8).

#### 3.4.1 State-Space and Dynamic Model

In order to model the dynamic behavior of the visible ends of MTs in our algorithm, we represent the object state with the state vector  $\mathbf{x}_t = (x_t, \dot{x}_t, y_t, \dot{y}_t, z_t, \dot{z}_t, \sigma_{\max,t}, \sigma_{\min,t}, \sigma_{z,t}, I_t)^T$ , where  $(\sigma_{\max,t}, \sigma_{\min,t}, \sigma_{z,t})^T \triangleq \mathbf{s}_t$  is the object shape feature vector (see Section 3.4.3),  $(x_t, y_t, z_t)^T \triangleq \mathbf{r}_t$  is the radius vector,  $\dot{\mathbf{r}}_t \triangleq \mathbf{v}_t$  is velocity, and  $I_t$  object intensity. The state evolution model  $D(\mathbf{x}_t|\mathbf{x}_{t-1})$  can be factorized as

$$D(\mathbf{x}_t|\mathbf{x}_{t-1}) = D_y(\mathbf{y}_t|\mathbf{y}_{t-1})D_s(\mathbf{s}_t|\mathbf{s}_{t-1})D_I(I_t|I_{t-1}), \quad (3.7)$$

where  $\mathbf{y}_t = (x_t, \dot{x}_t, y_t, \dot{y}_t, z_t, \dot{z}_t)$ . Here,  $D_y(\mathbf{y}_t|\mathbf{y}_{t-1})$  is modeled using a linear Gaussian model [40], which can easily be evaluated pointwise in (3.4), and is given by

$$D_y(\mathbf{y}_t|\mathbf{y}_{t-1}) \propto \exp\left(-\frac{1}{2}(\mathbf{y}_t - \mathbf{F}\mathbf{y}_{t-1})^T \mathbf{Q}^{-1}(\mathbf{y}_t - \mathbf{F}\mathbf{y}_{t-1})\right), \quad (3.8)$$

with the process transition matrix  $\mathbf{F} = \text{diag}[\mathbf{F}_1, \mathbf{F}_1, \mathbf{F}_1]$  and covariance matrix  $\mathbf{Q} = \text{diag}[\mathbf{Q}_1, \mathbf{Q}_1, \mathbf{Q}_1]$  given by

$$\mathbf{F}_1 = \begin{pmatrix} 1 & T \\ 0 & 1 \end{pmatrix} \quad \text{and} \quad \mathbf{Q}_1 = \begin{pmatrix} q_{11} & q_{12} \\ q_{12} & q_{22} \end{pmatrix},$$

where  $T$  is the sampling interval. Depending on the parameters  $q_{11}$ ,  $q_{12}$ ,  $q_{22}$  the model (3.8) describes a variety of motion patterns, ranging from random walk ( $\|\mathbf{v}_t\| = 0$ ,  $q_{11} \neq 0$ ,  $q_{12} = 0$ ,  $q_{22} = 0$ ) to nearly constant velocity ( $\|\mathbf{v}_t\| \neq 0$ ,  $q_{11} \neq 0$ ,  $q_{12} \neq 0$ ,  $q_{22} \neq 0$ ) [11], [84]. In our application, the parameters are fixed to  $q_{11} = \frac{q_1}{3}T^3$ ,  $q_{12} = \frac{q_1}{2}T^2$ ,  $q_{22} = q_1T$ , where  $q_1$  controls the noise level. In this case, model (3.8) corresponds to the continuous-time model  $\dot{\mathbf{r}}(t) = \mathbf{w}(t) \approx 0$ , where  $\mathbf{w}(t)$  is white

noise that corresponds to noisy accelerations [11]. We also make the realistic assumption that object velocities are bounded. This prior information is object dependent and will be used for state initialization (see Section 3.4.8). Small changes in frame-to-frame MT appearance (shape) are modeled using the Gaussian transition prior  $D_s(\mathbf{s}_t|\mathbf{s}_{t-1}) = \mathcal{N}(\mathbf{s}_t|\mathbf{s}_{t-1}, Tq_2\mathbf{I})$ , where  $\mathcal{N}(\cdot|\boldsymbol{\mu}, \boldsymbol{\Sigma})$  indicates the normal distribution with mean  $\boldsymbol{\mu}$  and covariance matrix  $\boldsymbol{\Sigma}$ ,  $\mathbf{I}$  is the identity matrix, and  $q_2$  represents the noise level in object appearance.

In practice, the analysis of time-lapse fluorescence microscopy images is complicated by photobleaching, a dynamic process by which the fluorescent proteins undergo photoinduced chemical destruction upon exposure to excitation light and thus lose their ability to fluoresce. Although the mechanisms of photobleaching are not yet well understood, two commonly used (and practically similar) approximations of fluorescence intensity over time are given by

$$I(t) = Ae^{-at} + B \quad (3.9)$$

and

$$I(t) = I_0 \left( 1 + \left( \frac{t}{L} \right)^k \right)^{-1}, \quad (3.10)$$

where  $A$ ,  $B$ ,  $a$ ,  $I_0$ ,  $L$ , and  $k$  are experimentally determined constants (see [124, 148] for more details on the validity and sensitivity of these models). The rate of photobleaching is a function of the excitation intensity. With a laser as an excitation source, photobleaching is observed on the time scale of microseconds to seconds. The high numerical aperture objectives currently in use, which maximize spatial resolution and improve the limits of detection, further accelerate the photobleaching process. Commonly, photobleaching is ignored by standard tracking methods, but in many practical cases it is necessary to model this process so as to be less sensitive to changing experimental conditions.

Following the common approximation (3.9), we model object intensity in our image data by the sum of a time-dependent, a time-independent, and a random component:

$$I_t + I_c + u_t = \frac{I_0\hat{A}}{\hat{A} + \hat{B}} e^{-\hat{\alpha}t} + \frac{I_0\hat{B}}{\hat{A} + \hat{B}} + u_t, \quad (3.11)$$

where  $u_t$  is zero-mean Gaussian process noise and  $I_0$  is the initial object intensity, obtained by the initialization procedure (see Section 3.4.8). The parameters  $\hat{A}$ ,  $\hat{B}$ , and  $\hat{\alpha}$  are estimated using the Levenberg-Marquardt algorithm for nonlinear fitting of (3.9) to the average background intensity over time,  $b_t$  (see Section 3.4.3). In order to conveniently incorporate the photobleaching effect contained in (3.11) into our framework, we approximate it as a first-order Gauss-Markov process,  $I_t = (1 - \hat{\alpha})I_{t-1} + u_t$ , which models the exponential intensity decay in the discrete-time domain. In this case, the corresponding state prior  $D_I(I_t|I_{t-1}) = \mathcal{N}(I_t|(1 - \hat{\alpha})I_{t-1}, q_3T)$ , where  $q_3 = T^{-1}\sigma_u^2$  and  $\sigma_u^2$  is the variance of  $u_t$ .

The photobleaching effect could alternatively be accommodated in our framework by assuming a constant intensity model ( $\hat{\alpha} = 0$ ) for  $D_I(I_t|I_{t-1})$ , but with a very high variance for the process noise,  $\sigma_u^2$ . However, in practice, because of the limited number



of MC samples, the variance of the estimation would rapidly grow, and many samples would be used inefficiently, causing problems especially in the case of a highly peaked likelihood  $L(\mathbf{z}_t|\mathbf{x}_t)$  (see Section 3.4.3). By using (3.11), we follow at least the trend of the intensity changes, and bring the estimation closer to the optimal solution. This way, we reduce the estimation variance and, consequently, the number of MC samples needed for the same accuracy as in the case of the constant intensity model.

In summary, the proposed model (3.7) correctly approximates small accelerations in object motion and fluctuations in object intensity, and therefore is very suitable for tracking growing MTs, as their dynamics can be well modeled by constant velocity plus small random diffusion [48]. The model (3.8) can also be successfully used for tracking other subcellular structures, for example vesicles, which are characterized by motion with higher nonlinearity. In that case, the process noise level, defined by  $\mathbf{Q}$ , should be increased.

### 3.4.2 Object Interactions and Markov Random Field

In order to obtain a more realistic motion model and avoid track coalescence in the case of multiple object tracking, we explicitly model the interaction between objects using a Markov random field (MRF) [76]. Here we use a pairwise MRF, expressed by means of a Gibbs distribution

$$\begin{aligned} \psi_t(\mathbf{x}_t^{(i)}, \mathbf{x}_t^{(j)}) &\propto \exp(-d_t^{i,j}), \\ i, j &\in \{1, \dots, N\}, \quad c_t^{(i)} \neq c_t^{(j)}, \end{aligned} \quad (3.12)$$

where  $d_t^{i,j}$  is a penalty function which penalizes the states of two objects  $c_t^{(i)}$  and  $c_t^{(j)}$  that are closely spaced at time  $t$ . That is,  $d_t^{i,j}$  is maximal when two objects coincide and gradually falls off as they move apart. This simple pairwise representation is easy to implement yet can be made quite sophisticated. Using this form, we can still retain the predictive motion model of each individual target. To this end, we sample  $N_s$  times the pairs  $(\mathbf{x}_{m,t-1}^{(l)}, \mathbf{x}_{m,t}^{(l)})$  ( $M$  such pairs at a time,  $m = \{1, \dots, M\}$ ), from  $p_m(\mathbf{x}_{t-1}|\mathbf{z}_{1:t-1})$  and  $q(\mathbf{x}_t|\mathbf{x}_{m,t-1}^{(l)}, \mathbf{z}_t)$ , respectively,  $l = \{1, \dots, N_s\}$ . Taking into account (3.12), the weights (3.4) in this case are given by

$$w_{m,t}^{(l)} \propto \frac{L(\mathbf{z}_t|\mathbf{x}_{m,t}^{(l)})D(\mathbf{x}_{m,t}^{(l)}|\mathbf{x}_{m,t-1}^{(l)})}{q(\mathbf{x}_{m,t}^{(l)}|\mathbf{x}_{m,t-1}^{(l)}, \mathbf{z}_t)} \prod_{k=1, k \neq m}^M \psi_t(\mathbf{x}_{m,t}^{(l)}, \mathbf{x}_{k,t}^{(l)}). \quad (3.13)$$

The mixture representation  $\{\{\mathbf{x}_{m,t}^{(l)}, w_{m,t}^{(l)}\}_{m=1}^M\}_{l=1}^{N_s}$  is then straightforwardly transformed to  $\{\mathbf{x}_t^{(i)}, w_t^{(i)}, c_t^{(i)}\}_{i=1}^N$ . In our application we have found that an interaction potential based only on object positions is sufficient to avoid most tracking failures. The use of a MRF approach is especially relevant and efficient in the case of 3D+ $t$  data analysis, because object merging is not possible in our application.

### 3.4.3 Observation Model and Likelihood

The measurements in our application are represented by a sequence of 2D or 3D images showing the motion of fluorescent proteins. The individual images (also called

frames) are recorded at discrete instants  $t$ , with a sampling interval  $T$ , with each image consisting of  $N_x \times N_y \times N_z$  pixels ( $N_z = 1$  in 2D). At each pixel  $(i, j, k)$ , which corresponds to a rectangular volume of dimensions  $\Delta_x \times \Delta_y \times \Delta_z$  nm<sup>3</sup>, the measured intensity is denoted as  $z_t(i, j, k)$ . The complete measurement recorded at time  $t$  is an  $N_x \times N_y \times N_z$  matrix denoted as  $\mathbf{z}_t = \{z_t(i, j, k) : i = 0, \dots, N_x - 1, j = 0, \dots, N_y - 1, k = 0, \dots, N_z - 1\}$ . For simplicity we assume that the origins and axis orientations of the  $(x, y, z)$  reference system and the  $(i, j, k)$  system coincide. Let  $\tilde{z}_t(\mathbf{r})$  denote a first-order interpolation of  $z_t(\Delta_x i, \Delta_y j, \Delta_z k)$ .

The image formation process in a microscope can be modeled as a convolution of the true light distribution coming from the specimen, with a point-spread function (PSF), which is the output of the optical system for an input point light source. The theoretical diffraction-limited PSF in the case of paraxial and non-paraxial imaging can be expressed by the scalar Debye diffraction integral [190]. In practice, however, a 3D Gaussian approximation of the PSF [161] is commonly favored over the more complicated PSF models (such as the Gibson-Lanni model [55]). This choice is mainly motivated by computational considerations, but a Gaussian approximation of the physical PSF is fairly accurate for reasonably large pinhole sizes (relative squared error (RSE)  $< 9\%$ ) and nearly perfect for typical pinhole sizes (RSE  $< 1\%$ ) [190]. In most microscopes currently used, the PSF limits the spatial resolution to  $\approx 200$  nm in-plane and  $\approx 600$  nm in the direction of the optical axis, as a consequence of which subcellular structures (typically of size  $< 20$  nm) are imaged as blurred spots. We adopt the common assumption that all blurring processes are due to a linear and spatially invariant PSF.

The PF framework accommodates any PSF that can be calculated pointwise. To model the imaged intensity profile of the object with some shape, one would have to use the convolution with the PSF for every state  $\mathbf{x}_t^{(i)}$ . In order to overcome this computational overload, we propose to model the PSF and object shape at the same time using the 3D Gaussian approximation. To model the manifest elongation in the intensity profile of MTs, we utilize the velocity components from the state vector  $\mathbf{x}_t$  as parameters in the PSF. In this case, for an object of intensity  $I_t$  at position  $\mathbf{r}_t$ , the intensity contribution to pixel  $(i, j, k)$  is approximated as

$$h_t(i, j, k; \mathbf{x}_t) = b_t + (I_t + I_c) \times \exp\left(-\frac{1}{2} \mathbf{m}^T \mathbf{R}^T \mathbf{\Sigma}^{-1} \mathbf{R} \mathbf{m}\right) \times \exp\left(-\frac{(k\Delta_z - z_t \|\mathbf{m}\| \tan \theta)^2}{2\sigma_z^2}\right), \quad (3.14)$$

where  $b_t$  is the background intensity,  $\sigma_z$  ( $\approx 235$  nm) models the axial blurring,  $\mathbf{R} = \mathbf{R}(\phi)$  is a rotation matrix

$$\mathbf{R}(\phi) = \begin{pmatrix} \cos \phi & \sin \phi \\ -\sin \phi & \cos \phi \end{pmatrix}, \quad \mathbf{\Sigma} = \begin{pmatrix} \sigma_m^2(\theta) & 0 \\ 0 & \sigma_{\min}^2 \end{pmatrix},$$

$$\mathbf{m} = \begin{pmatrix} i\Delta_x - x_t \\ j\Delta_y - y_t \end{pmatrix}, \quad \sigma_m(\theta) = \sigma_{\min} - (\sigma_{\min} - \sigma_{\max}) \cos \theta,$$

$$\tan \theta = \frac{\dot{z}_t}{\sqrt{\dot{x}_t^2 + \dot{y}_t^2}}, \quad \tan \phi = \frac{\dot{y}_t}{\dot{x}_t}, \quad -\pi < \phi, \theta \leq \pi.$$

The parameters  $\sigma_{\max}$  and  $\sigma_{\min}$  represent the amount of blurring and, at the same time, model the elongation of the object along the direction of motion. For subresolution structures such as vesicles,  $\sigma_{\min} = \sigma_{\max} \approx 80$  nm, and for the elongated MTs  $\sigma_{\min} \approx 100$  nm and  $\sigma_{\max} \approx 300$  nm.

For background level estimation we use the fact that the contribution of object intensity values to the total image intensity (mainly formed by background structures with lower intensity) is negligible, especially in the case of low SNRs. We have found that in a typical 2D image of size  $10^3 \times 10^3$  pixels containing a thousand objects, the number of object pixels is only about 1%. Even if the object intensities would be 10 times as large as the background level (very high SNR), their contribution to the total image intensity would be less than 10%. In that case, the normalized histogram of the image  $\mathbf{z}_t$  can be approximated by a Gaussian distribution with mean  $\hat{b}$  and variance  $\sigma_b^2$ . The estimated background  $b_t = \hat{b}$  is then calculated according to

$$b_t = \frac{1}{N_x N_y N_z} \sum_{i=0}^{N_x-1} \sum_{j=0}^{N_y-1} \sum_{k=0}^{N_z-1} z_t(i, j, k). \quad (3.15)$$

In the case of a skewed histogram of image intensity, the median of the distribution can be taken as an estimate of the background level. The latter is preferable because it treats object pixels as outliers for the background distribution.

Since an object will affect only the pixels in the vicinity of its location,  $\mathbf{r}_t$ , we define the likelihood function as

$$L_G(\mathbf{z}_t | \mathbf{x}_t) \triangleq \prod_{(i,j,k) \in C(\mathbf{x}_t)} \frac{p_h(z_t(i, j, k) | \mathbf{x}_t)}{p_b(z_t(i, j, k) | b_t)}, \quad (3.16)$$

where  $C(\mathbf{x}_t) = \{(i, j, k) \in \mathbb{Z}^3 : h_t(i, j, k; \mathbf{x}_t) - b_t > 0.1I_t\}$ ,

$$p_h(z_t(i, j, k) | \mathbf{x}_t) \propto \frac{1}{\sigma_h(i, j, k)} \exp \left( -\frac{(z_t(i, j, k) - h_t(i, j, k; \mathbf{x}_t))^2}{2\sigma_h^2(i, j, k)} \right), \quad (3.17)$$

and

$$p_b(z_t(i, j, k) | b_t) \propto \exp \left( -\frac{(z_t(i, j, k) - b_t)^2}{2\sigma_b^2} \right), \quad (3.18)$$

with  $\sigma_h^2(i, j, k)$  and  $\sigma_b^2$  the variances of the measurement noise for the object + background and background, respectively, which are assumed to be independent from pixel to pixel and from frame to frame. Poisson noise, which can be used to model the effect of the quantum nature of light on the measured data, is one of the main sources of noise in fluorescence microscopy imaging. The recursive Bayesian solution is applicable as long as the statistics of the measurement noise is known for each pixel. In this chapter we use a valid approximation of Poisson noise, with  $\sigma_h^2(i, j, k) = h_t(i, j, k; \mathbf{x}_t)$  and  $\sigma_b^2 = b_t$ , by scaling the image intensities in order to satisfy the condition  $\sigma_b^2 = b_t$  [26].

### 3.4.4 Hierarchical Searching

Generally, the likelihood  $L_G(\mathbf{z}_t|\mathbf{x}_t)$  is very peaked (even when the region  $C(\mathbf{x}_t)$  is small) and may lead to severe sample impoverishment and divergence of the filter. Theoretically it is impossible to avoid the degeneracy phenomenon, where, after a few iterations of the algorithm, all but one of the normalized importance weights are very close to zero [40]. Consequently, the accuracy of the estimator also degrades enormously [125]. A commonly used measure of degeneracy is the *estimated effective sample size* [40], given by

$$N_{eff}(t) = \left( \sum_{i=1}^{N_s} (w_t^{(i)})^2 \right)^{-1}, \quad (3.19)$$

which intuitively corresponds to the number of “useful” particles. Degeneracy is usually strong for image data with low SNR, but the filter also performs poorly when the noise level is too small [39]. This suggests that MC estimation with accurate sensors may perform worse than with inaccurate sensors. The problem can be partially fixed by using an observation model which overestimates the measurement noise. While the performance is better, this is not a principled way of fixing the problem; the observation model is artificially inaccurate and the resulting estimation is no longer a posterior, even if infinitely many samples were used. Other methods that try to improve the performance of PF include partitioned sampling [89], the auxiliary particle filter (APF) [9], [126] and the regularized particle filters (RPF) [39, 126]. Because of the highly nonlinear observation model and dynamic model with a high noise level, the mentioned methods are inefficient for our application. Partitioned sampling requires the possibility to partition the state space and to decouple the observation model for each of the partitions, which cannot be done for our application. Application of the APF is beneficial only when the dynamic model is correctly specified with a small amount of process noise. The tracking of highly dynamic structures with linear models requires increasing the process noise in order to capture the typical motion patterns.

To overcome these problems, we use a different approach, based on RPF, and mainly on progressive correction [39]. First, we propose a second observation model:

$$L_S(\mathbf{z}_t|\mathbf{x}_t) \triangleq \frac{\sigma_B}{\sigma_S(\mathbf{x}_t)} \exp \left( \frac{(S_t^z(\mathbf{x}_t) - S_t^b(\mathbf{x}_t))^2}{2\sigma_B^2} - \frac{(S_t^z(\mathbf{x}_t) - S_t^h(\mathbf{x}_t))^2}{2\sigma_S^2(\mathbf{x}_t)} \right), \quad (3.20)$$

where

$$S_t^z(\mathbf{x}_t) = \sum_{(i,j,k) \in C(\mathbf{x}_t)} z_t(i, j, k),$$

and

$$S_t^h(\mathbf{x}_t) = \sum_{(i,j,k) \in C(\mathbf{x}_t)} h_t(i, j, k; \mathbf{x}_t),$$

$S_t^b = b_t |C(\mathbf{x}_t)|$ , where  $|\cdot|$  denotes the set size operator, and the variances  $\sigma_S^2$  and  $\sigma_B^2$  are taken to approximate the Poisson distribution:  $\sigma_S^2 = S_t^o$  and  $\sigma_B^2 = S_t^b$ . The likelihood  $L_S(\mathbf{z}_t|\mathbf{x}_t)$  is less peaked but gives an error of the same order as  $L_G(\mathbf{z}_t|\mathbf{x}_t)$ . Another advantage is that  $L_S(\mathbf{z}_t|\mathbf{x}_t)$  can be used for objects without a predefined

shape; only the region  $C(\mathbf{x}_t)$ , which presumably contains the object, and the total object intensity in  $C(\mathbf{x}_t)$  need to be specified.

Subsequently, we propose a modified hierarchical search strategy, which uses both models,  $L_S$  and  $L_G$ . To this end, we calculate an intermediate state at time  $t'$ , between time points  $t-1$  and  $t$ , by propagating and updating the samples using the likelihood  $L_S$  according to

$$\bar{p}(\mathbf{x}_{t'}|\mathbf{z}_{1:t'}) \propto L_S(\mathbf{z}_{t'}|\mathbf{x}_{t'})D(\mathbf{x}_{t'}|\mathbf{x}_{t-1})p(\mathbf{x}_{t-1}|\mathbf{z}_{1:t-1}) \quad (3.21)$$

where  $\mathbf{z}_{t'} = \mathbf{z}_t$ . After this step,  $N_{eff}$  is still rather high, because the likelihood  $L_S$  is less peaked than  $L_G$ . In a next step, particles with high weights at time  $t'$  are diversified and put into regions where the likelihood  $L_G$  is high, giving a much better approximation of the posterior:

$$p(\mathbf{x}_t|\mathbf{z}_{1:t}) \propto L_G(\mathbf{z}_t|\mathbf{x}_t)\mathcal{N}(\mathbf{x}_t|\boldsymbol{\mu}_{t'}, \boldsymbol{\Sigma}_{t'})\bar{p}(\mathbf{x}_{t'}|\mathbf{z}_{1:t'}), \quad (3.22)$$

where the expectation and the variance are given by

$$\boldsymbol{\mu}_{t'} = \mathbb{E}_{\bar{p}}[\mathbf{x}_{t'}], \quad \boldsymbol{\Sigma}_{t'} = \mathbb{E}_{\bar{p}}[(\mathbf{x}_{t'} - \boldsymbol{\mu}_{t'})(\mathbf{x}_{t'} - \boldsymbol{\mu}_{t'})^T]. \quad (3.23)$$

The described hierarchical search strategy is further denoted as  $L_{SG}$ . It keeps the number  $N_{eff}$  quite large and, in practice, provides filters that are more stable in time, with lower variance in the position estimation.

### 3.4.5 Measurement Gating

Multiple object tracking requires gating, or measurement selection. The purpose of gating is to reduce computational expense by eliminating measurements which are far from the predicted measurement location. Gating is performed for each track at each time step  $t$  by defining a subvolume of the image space, called the gate. All measurements positioned within the gate are selected and used for the track update step, (3.2), while measurements outside the gate are ignored in these computations. In standard approaches to tracking, using the Kalman filter or extended Kalman filter, measurement gating is accomplished by using the predicted measurement covariance for each object and then updating the predicted state using joint probabilistic data association [79]. In the PF approach, which is able to cope with nonlinear and non-Gaussian models, the analog of the predicted measurement covariance is not available and can be constructed only by taking, for example, a Gaussian approximation of the current particle cloud and using it to perform gating. Generally, this approximation is unsatisfactory, since the advantages gained from having a representation of a non-Gaussian pdf are lost. In the proposed framework, however, this approximation is justified by using the highly peaked likelihood functions and the reclustering procedure (described in Section 3.4.7), which keep the mixture components unimodal.

Having the measurements  $\tilde{z}_t(\mathbf{r}_t)$ , we define the gate for each of the tracks as follows:

$$C_{m,t} = \{\mathbf{r}_t \in \mathbb{R}^3 : (\mathbf{r}_t - \bar{\mathbf{r}}_{m,t})^T \boldsymbol{\Sigma}_{m,t}^{-1} (\mathbf{r}_t - \bar{\mathbf{r}}_{m,t}) \leq C_0\}, \quad (3.24)$$

where the parameter  $C_0$  specifies the size of the gate, which is proportional to the probability that the object falls within the gate. Generally, since the volume of the

gate is dependent on the tracking accuracy, it varies from scan to scan and from track to track. In our experiments,  $C_0 = 9$  (a 3-standard-deviation level gate). The gate  $C_{m,t}$  is centered at the position predicted from the particle representation of  $p_m(\mathbf{x}_t|\mathbf{z}_{1:t-1})$ :

$$\bar{\mathbf{r}}_{m,t} = \mathbb{E}_{p_m}[\mathbf{r}_t] = \int \mathbf{r}_t p_m(\mathbf{x}_t|\mathbf{z}_{1:t-1}) d\mathbf{x}_t \approx \sum_{\substack{i=1, \\ c_{t-1}^{(i)}=m}}^N \bar{\mathbf{r}}_t^{(i)} w_{t-1}^{(i)}, \quad (3.25)$$

where the  $\bar{\mathbf{r}}_t^{(i)}$  are the position elements of the state vector

$$\bar{\mathbf{x}}_t^{(i)} \sim D(\mathbf{x}_t|\mathbf{x}_{t-1}^{(i)}), \quad i = \{1, \dots, N\}.$$

Similarly, the covariance matrix is calculated as

$$\Sigma_{m,t} = \mathbb{E}_{p_m}[(\mathbf{r}_t - \bar{\mathbf{r}}_{m,t})(\mathbf{r}_t - \bar{\mathbf{r}}_{m,t})^T]. \quad (3.26)$$

### 3.4.6 Data-Dependent Sampling

Basic particle filters [9, 70, 143], which use the proposal distribution  $q(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{z}_t) = D(\mathbf{x}_t|\mathbf{x}_{t-1})$  usually perform poorly because too few samples are generated in regions where the desired posterior  $p(\mathbf{x}_t|\mathbf{z}_{1:t})$  is large. In order to construct a proposal distribution which alleviates this problem and takes into account the most recent measurements  $\mathbf{z}_t$ , we propose to transform the image sequence into probability distributions. True spots are characterized by a combination of convex intensity distributions and a relatively high intensity. Noise-induced local maxima typically exhibit a random distribution of intensity changes in all directions, leading to a low local curvature [161]. These two discriminative features (intensity and curvature) are used to construct an approximation of the likelihood  $L(\mathbf{z}_t|\mathbf{x}_t)$ , using the image data available at time  $t$ . For each object we use the transformation

$$\tilde{p}_m(\mathbf{r}_t|\mathbf{z}_t) = \frac{(G_\sigma * \tilde{z}_t(\mathbf{r}_t) - b_t)^r \kappa_t^s(\mathbf{r}_t)}{\int_{C_{m,t}} (G_\sigma * \tilde{z}_t(\mathbf{r}_t) - b_t)^r \kappa_t^s(\mathbf{r}_t) dx dy dz}, \quad (3.27)$$

$\forall \mathbf{r}_t \in C_{m,t}$ , where  $G_\sigma$  is the Gaussian kernel with standard deviation (scale)  $\sigma$ , the curvature  $\kappa_t(\mathbf{r}_t)$  is given by the determinant of the Hessian matrix  $\mathbf{H}$  of the intensity  $\tilde{z}_t(\mathbf{r}_t)$ :

$$\kappa_t(\mathbf{r}_t) = \det(\mathbf{H}(\mathbf{r}_t)), \quad \mathbf{H}(\mathbf{r}_t) = \nabla \cdot \nabla^T \tilde{z}_t(\mathbf{r}_t), \quad (3.28)$$

and the exponents  $r > 0$  and  $s > 0$  weigh each of the features and determine the peakedness of the likelihood.

Using this transformation, we define the new data dependent proposal distribution for object  $m$  as

$$\begin{aligned} \tilde{q}_m(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{z}_t) = & \tilde{p}_m(\mathbf{r}_t|\mathbf{z}_t) \mathcal{N}(I_t|\tilde{z}_t(\mathbf{r}_t) - b_t, q_3 T) \times \\ & \mathcal{N}(\mathbf{s}_t|\mathbf{s}_{m,t-1}^{\text{MMSE}}, T q_2 \mathbf{I}) \mathcal{N}(\mathbf{v}_t|\mathbf{r}_t - \hat{\mathbf{r}}_{m,t-1}^{\text{MMSE}}, T q_1 \mathbf{I}), \end{aligned} \quad (3.29)$$

Contrary to the original proposal distribution, which fails if the likelihood is too peaked, the distribution (3.29) generates samples that are highly consistent with the most recent measurements in the predicted (using the information from the previous time step) gates. A combination of both proposal distributions gives excellent results:

$$q_m(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{z}_t) = \gamma D(\mathbf{x}_t|\mathbf{x}_{t-1}) + (1 - \gamma)\tilde{q}_m(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{z}_t),$$

where  $0 < \gamma < 1$ . Comparison shows that the proposal distribution  $q_m(\mathbf{x}_t|\mathbf{x}_{t-1}, \mathbf{z}_t)$  is uniformly superior to the regular one ( $\gamma = 1$ ) and scales much better to smaller sample sizes.

### 3.4.7 Clustering and Track Management

The representation of the filtering distribution  $p(\mathbf{x}_t|\mathbf{z}_{1:t})$  as the mixture model (3.6) allows for a deterministic spatial reclustering procedure  $(\{c_t^{(i)}\}, M') = F(\{\mathbf{x}_t^{(i)}\}, \{c_t^{(i)}\}, M)$  [174]. The function  $F$  can be implemented in any convenient way. It calculates a new mixture representation (with possibly a different number of mixture components) taking as input the current mixture representation. This allows modeling and capturing merging and splitting events, which also have a direct analogy with biological phenomena. In our implementation, at each iteration the mixture representation is recalculated by applying  $K$ -means clustering algorithm. The reclustering is based on spatial information (object positions) only and is initialized with the estimates (3.25).

Taking into account our application, two objects are not allowed to merge when their states become similar. Whenever objects pass close to one another, the object with the best likelihood score typically “hijacks” the particles of the nearby mixture components. As mentioned above, this problem is partly solved by using the MRF model for object interactions. The MRF model significantly improves the tracking performance in 3D+t. For 2D+t data sets, however, the observed motion is a projection of the real 3D motion onto the 2D plane. In this case, when one object passes above or beneath another (in 3D), we perceive the motion as penetration or merging. These situations are in principle ambiguous and frequently cannot be resolved uniquely, neither by an automatic tracking method nor by a human observer.

We detect possible object intersections during tracking by checking whether the gates  $C_{m,t}$  intersect each other. For example, for two trajectories, the intersection is captured if  $C_{i,t} \cap C_{j,t} \neq \{0\}$ ,  $i, j \in \{1, \dots, M\}$ . In general, the measurement space  $C_t = \cup_{m=1}^M C_{m,t}$  is partitioned into a set of disjoint regions  $C_t = \{C_{1,t}^*, \dots, C_{K,t}^*\}$ , where  $C_{k,t}^*$  is either the union of connected gates or the gate itself. For each  $C_{k,t}^*$ , we define a set of indices  $J_{k,t}$ , which indicate which of the gates  $C_{i,t}$  belong to it:

$$J_{k,t} = \{i \in \{1, \dots, M\} : C_{i,t} \in C_{k,t}^*\} \quad (3.30)$$

For the gates  $C_{k,t}^*$  with  $|J_{k,t}| = 1$ , the update of the MC weights  $w_{m,t}^{(i)}$  is done according to (3.4). For all other gates  $C_{k,t}^*$ , which correspond to object interaction, we follow the procedure similar to the one described in Section 3.4.2. For each  $C_{k,t}^*$  for which  $|J_{k,t}| \neq 1$ , the set of states  $\{\mathbf{x}_{j,t}^{(l)}\}$ ,  $j \in J_{k,t}$ , is sampled from the proposal distribution (for every  $l = \{1, \dots, N_s\}$ ), and a set of hypotheses  $\Theta_{k,t}^{(l)} = \{\theta_1^{(l)}, \dots, \theta_S^{(l)}\}$ ,  $S = 2^{|J_{k,t}|}$ ,



is formed. Each  $\theta_i^{(l)}$  is a set of binary associations,  $\{a_{i,j}^{(l)}\}$ ,  $j \in J_{k,t}$ , where  $a_{i,j}^{(l)} = 1$  if object  $j$  exists during the interaction, and  $a_{i,j}^{(l)} = 0$  if the object “dies” or leaves just before or during the interaction and gives no measurements at time  $t$ . The hypothesis that maximizes the likelihood is selected as

$$\hat{\theta}_k^{(l)} = \operatorname{argmax}_{\theta_i^{(l)} \in \Theta_{k,t}^{(l)}} L(\mathbf{z}_t | \mathbf{x}_t), \quad (3.31)$$

where the likelihood  $L(\mathbf{z}_t | \mathbf{x}_t)$  can be either  $L_G(\mathbf{z}_t | \mathbf{x}_t)$  or  $L_S(\mathbf{z}_t | \mathbf{x}_t)$ , but the region  $C(\mathbf{x}_t)$  is defined as  $C(\mathbf{x}_t) = \cup_{j \in J_{k,t}} C(\mathbf{x}_{j,t}^{(l)})$ , and  $h_t(\cdot; \mathbf{x}_t)$  is substituted in (3.16) and (3.20) for each  $\theta_i^{(l)}$  with  $\sum_{j \in J_{k,t}} a_{i,j}^{(l)} h_t(\cdot; \mathbf{x}_{j,t}^{(l)})$ . For the update of the MC weights  $w_{j,t}^{(l)}$  the region  $C(\mathbf{x}_t) = C(\mathbf{x}_{j,t}^{(l)})$  and  $h_t(\cdot; \mathbf{x}_t) = \sum_{j \in J_{k,t}} \hat{a}_j^{(l)} h_t(\cdot; \mathbf{x}_{j,t}^{(l)})$  are used in (3.16) and (3.20), with the  $\hat{a}_j^{(l)}$  denoting the  $a_{i,j}^{(l)}$  corresponding to  $\hat{\theta}_k^{(l)}$ . Additionally, in such cases, we do not perform reclustering, but keep the labels for the current iteration as they were before. If the component representation in the next few frames after the interaction event becomes too diffuse, and there is more than one significant mode, splitting is performed and a new track is initiated (see Section 3.4.8 for more details).

Finally, for the termination of an existing track, the methods commonly used for small target tracking [68, 104] cannot be applied straightforwardly. These methods assume that, due to imperfect sensors, the probability of detecting an object is less than one, and they try to follow the object after disappearance for 4-5 frames, predicting its position in time and hoping to catch it again. In our case, when the density of objects in the images is high, such monitoring would definitely result in “confirming” measurements after 3-5 frames of prediction, but these measurements would very likely originate from another object. In our algorithm in order to terminate the track we define the thresholds  $\bar{\sigma}_{\max}$ ,  $\bar{\sigma}_{\min}$ ,  $\bar{\sigma}_z$  that describe the “biggest” objects that we are going to track. Then we sample the particles in the predicted gates  $C_{m,t}$  using the data-dependent sampling (3.27) with  $s = 0$ . If the determinant of the covariance matrix computed for those MC samples is greater than  $\bar{\sigma}_{\max}^2 \bar{\sigma}_{\min}^2 \bar{\sigma}_z^2 r^{-3}$  the track is terminated. If the gate  $C_{m,t}$  does not contain a real object the determinant value will be much higher than the proposed threshold, which is nicely separate the objects from the background structures.

### 3.4.8 Initialization and Track Initiation

The prior distribution  $p(\mathbf{x}_0)$  is specified based on information available in the first frame. One way to initialize the state vector  $\mathbf{x}_0$  would be to point on the desired bright spots in the image or to select regions of interest. In the latter case, the state vector is initialized by a uniform distribution over the state space, in predefined intervals for velocity and intensity, and the expected number of objects should be specified. During filtering and reclustering, after a burn-off period of 2-3 frames, only the true objects will remain.

For completely automatic initiation of object tracks in the first frame, and also for the detection of potential objects for tracking in subsequent frames, we use the following procedure. First, the image space is divided into  $N_I = N_X \times N_Y \times N_Z$

rectangular 3D cells of dimensions  $\Delta_c \times \Delta_c \times \Delta_a$ , with  $\Delta_c = 6\sigma_{\max}$  and  $\Delta_a = 6\sigma_z$ . Next, for each time step  $t$ , the image is converted to a probability map according to (3.27), and  $N = MN_s$  particles  $\tilde{\mathbf{x}}_t^{(i)}$  are sampled with equal weights. The number of particles in each cell represents the degree of belief in object birth. To discriminate potential objects from background structures or noise, we estimate for each cell the center of mass  $\hat{\mathbf{r}}_k$  ( $k = \{1, \dots, N_I\}$ ) by MC integration over that cell and calculate the number of MC samples  $n_{k,t}$  in the ellipsoidal regions  $S_{k,t}(\mathbf{r}_t)$  centered at  $\hat{\mathbf{r}}_k$  (with semi-axes of lengths  $\Delta_c/2$ ,  $\Delta_c/2$ ,  $\Delta_a/2$ ). In order to initiate a new object, two conditions have to be satisfied. The first condition is that  $n_{k,t}$  should be greater than  $N \frac{|S_{k,t}|}{|\mathbf{z}_t|} = N\pi(6N_I)^{-1}$ . The threshold represents the expected number of particles if the sampling was done from the image region with uniform background intensity. The second condition is similar to the one for track termination (see Section 3.4.7): the determinant of the covariance matrix should be smaller than  $\bar{\sigma}_{\max}^2 \bar{\sigma}_{\min}^2 \bar{\sigma}_z^2 r^{-3}$ .

Each object  $d$  (out of  $M_d$  newly detected at time  $t$ ) is initialized with mixture weight  $\pi_{d,t} = (M + M_d)^{-1}$  and object position  $\mathbf{r}_{d,t}$  (the center of mass calculated by MC integration over the region  $S_{d,t}(\mathbf{r}_t)$ ). The velocity is uniformly distributed in a predefined range and the intensity is obtained from the image data for that frame and position. In cases where the samples from an undetected object are split between four cells (in the unlikely event when the object is positioned exactly on the intersection of the cell borders), the object will most probably be detected in the next time frame.

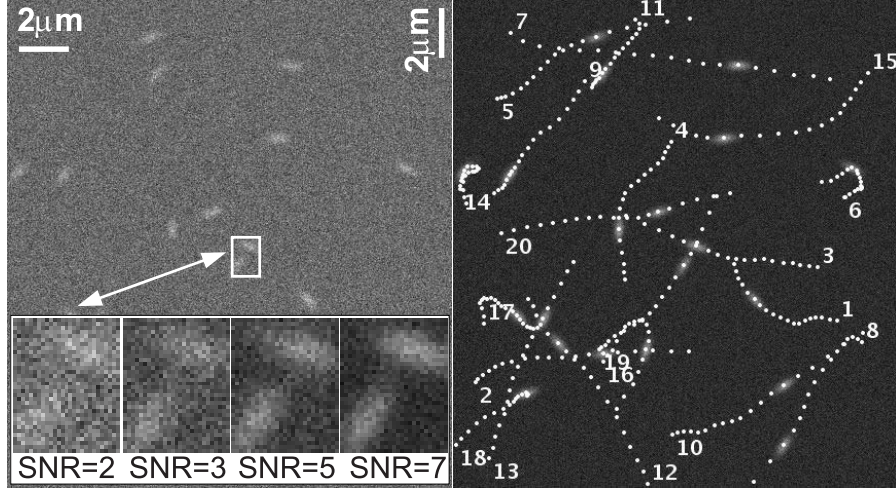
## 3.5 Experimental Results

The performance of the described PF-based tracking method was evaluated using both computer generated image data (Section 3.5.1) and real fluorescence microscopy image data from MT dynamics studies (Section 3.5.2). The former allowed us to test the accuracy and robustness to noise and object interaction of our algorithm compared to two other commonly used tracking tools. The experiments on real data enabled us to compare our algorithm to expert human observers.

### 3.5.1 Evaluation on Synthetic Data

#### 3.5.1.1 Simulation Setup

The algorithm was evaluated using synthetic but realistic 2D image sequences (20 time frames of  $512 \times 512$  pixels,  $\Delta_x = \Delta_y = 50$  nm,  $T = 1$  sec) of moving MT-like objects (a fixed number of 10, 20, or 40 objects per sequence, yielding data sets of different object densities), generated according to (3.8) and (3.14), for different levels of Poisson noise (see Fig. 3.2) in the range SNR=2–7, since SNR=4 has been identified by previous studies [26, 32] as a critical level at which several popular tracking methods break down. In addition, the algorithm was tested using 3D synthetic image sequences (20 time frames of  $512 \times 512$  pixels  $\times$  20 optical slices,  $\Delta_x = \Delta_y = 50$  nm,  $\Delta_z = 200$  nm,  $T = 1$  sec, with 10–40 objects per sequence), also for different noise levels in the range of SNR=2–7. Here, SNR is defined as the difference in intensity between the object and the background, divided by the standard deviation of the object noise [32]. The



**Figure 3.2.** Examples of synthetic images used in the experiments. The left image is a single frame from one of the sequences, at SNR=2, giving an impression of object appearance. The insets show zooms of objects at different SNRs. The right image is a frame from another sequence, at SNR=7, with the trajectories of the 20 moving objects superimposed (white dots), illustrating the motion patterns allowed by the linear state evolution model (3.8).

velocities of the objects ranged from 200 to 700 nm/sec, representative of published data [155].

Having the ground truth for the synthetic data, we evaluated the accuracy of tracking by using a traditional quantitative performance measure: the root mean square error (RMSE), in  $K$  independent runs (we used  $K = 3$ ) [104]:

$$\text{RMSE} = \sqrt{\frac{1}{K} \sum_{i=1}^K \text{RMSE}_k^2}, \quad (3.32)$$

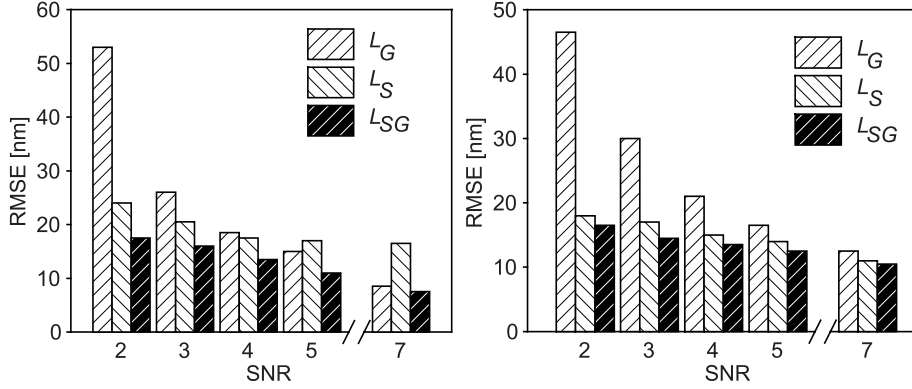
with

$$\text{RMSE}_k^2 = \frac{1}{M} \sum_{m=1}^M \left\{ \frac{1}{|\mathcal{T}_m|} \sum_{t \in \mathcal{T}_m} \|\mathbf{r}_{m,t} - \hat{\mathbf{r}}_{m,t}^k\|^2 \right\}, \quad (3.33)$$

where  $\mathbf{r}_{m,t}$  defines the true position of object  $m$  at time  $t$ ,  $\hat{\mathbf{r}}_{m,t}^k$  is a posterior mean estimate of  $\mathbf{r}_{m,t}$  for the  $k$ th run, and  $\mathcal{T}_m$  is the set of time points at which object  $m$  exists.

### 3.5.1.2 Experiments with Hierarchical Searching

In order to show the advantage of using the proposed hierarchical search strategy (see Section 3.4.4), we calculated the localization error at different SNRs for objects moving along horizontal straight lines at a constant speed of 400 nm/sec (similar



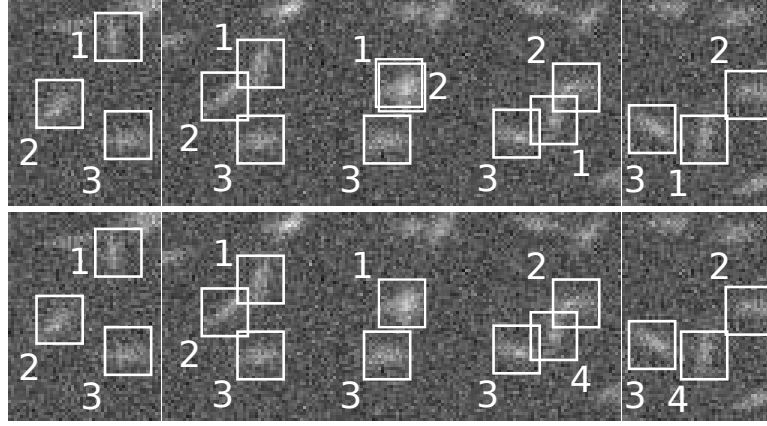
**Figure 3.3.** The RMSE in object position estimation as a function of SNR for round (left) and elongated (right) objects using the three different observation models,  $L_G$ ,  $L_S$ , and  $L_{SG}$ .

to [132]). The tracking was done for two types of objects: round ( $\sigma_{\max} = \sigma_{\min} = 100$  nm) and elongated ( $\sigma_{\max} = 300$  nm,  $\sigma_{\min} = 100$  nm) using the likelihoods  $L_S$ ,  $L_G$ , and the combined two-step approach  $L_{SG}$ . The filtering was performed with 500 MC samples. The RMSE for all three models is shown in Fig. 3.3. The localization error of the hierarchical search is lower and the effective sample size  $N_{eff}$  is higher than in the case of using only  $L_G$ . For comparison, for the likelihoods  $L_S$ ,  $L_G$ , and  $L_{SG}$ , the ratios between the effective sample size  $N_{eff}$  and  $N_s$  are less than 0.5, 0.005, and 0.05, respectively.

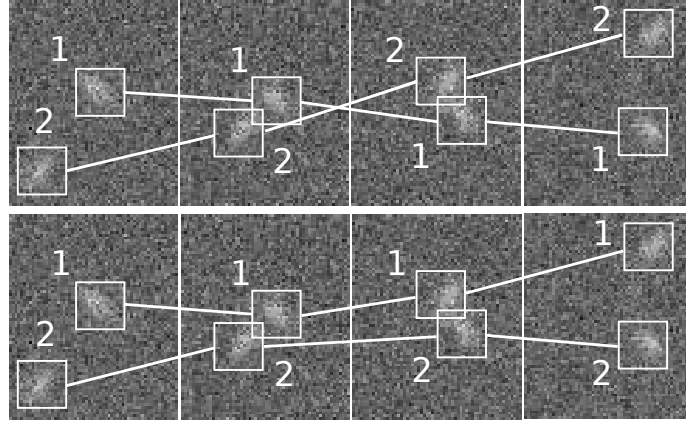
### 3.5.1.3 Comparison with Conventional Two-Stage Tracking Methods

The proposed PF-based tracking method was compared to conventional two-stage (completely separated detection and linking) tracking approaches commonly found in the literature. To maximize the credibility of these experiments, we chose to use two existing, state-of-the-art multitarget tracking software tools based on this principle, rather than making our own (possibly biased) implementation of described methods. The first is *Volocity* (Improvision, Coventry, UK), which is a commercial software package, and the second is *ParticleTracker* [132], which is freely available as a plugin to the public-domain image analysis tool ImageJ [121] (National Institutes of Health, Bethesda, MD, USA).

With *Volocity*, the user has to specify thresholds for the object intensity and the approximate object size in order to discriminate objects from the background, in the detection stage. These thresholds are set globally, for the entire image sequence. Following the extraction of all objects in each frame, linking is performed on the basis of finding nearest neighbors in subsequent image frames. This association of nearest neighbors also takes into account whether the motion is smooth or erratic. With *ParticleTracker*, the detection part also requires setting intensity and

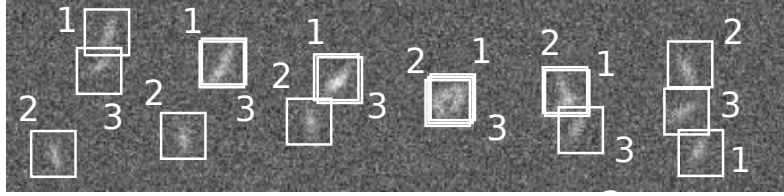


**Figure 3.4.** Example (SNR=3) showing the ability of our PF method to deal with one-frame occlusion scenarios (top sequence), using the proposed reclustering procedure, while *ParticleTracker* (and similarly *Volocity*) fails (bottom sequence).



**Figure 3.5.** Typical example (SNR=3) showing the ability of our PF method to resolve object crossing correctly (top sequence), by using the information about the object shape during the measurement-to-track association process, while *ParticleTracker* (and similarly *Volocity*) fails (bottom sequence).

object size thresholds. The linking, however, is based on finding the global optimal solution for the correspondence problem in a given number of successive frames. The solution is obtained using graph theory and global energy minimization [132]. The linking also utilizes the zeroth- and second-order intensity moments of the object intensities. This better resolves intersection problems and improves the linking result. For both tools, the parameters were optimized manually during each stage, until all objects in the scene were detected. Our PF-based method was initialized using the automatic initialization procedure described in Section 3.4.8. The user-definable algorithm parameters were fixed to the following values:  $\sigma_{\max} = 250$  nm,  $\sigma_{\min} = 120$  nm,



**Figure 3.6.** Example (SNR=3) where our PF method as well as *ParticleTracker* and *Volocity* failed (only the true tracks are shown in the sequence), because three objects interact at one location and the occlusion lasts for more than one frame.

**Table 3.1.** Comparison of the ability of the three methods to track objects correctly in cases of object appearance, disappearance, and interactions.

SNR	Volocity		ParticleTracker		Particle Filter	
	$r_0$	$r_1$	$r_0$	$r_1$	$r_0$	$r_1$
	$N_{tr} = 10$					
2	1.1	0.9	1.8	0.1	1	1
3	1	1	1	0.5	1	1
4	1	1	1	0.7	1	1
5	1	1	1	1	1	1
7	1	1	1	1	1	1
	$N_{tr} = 20$					
2	1.15	0.5	2	0.1	1.05	0.8
3	1.05	0.6	1.95	0.15	1	0.9
4	1.05	0.6	1.35	0.45	1	0.95
5	1	0.7	1.1	0.65	1	1
7	1	0.85	1.05	0.9	1	1
	$N_{tr} = 40$					
2	1.9	0.05	1.7	0.1	1.05	0.5
3	1.1	0.6	1.5	0.15	1.02	0.7
4	1.05	0.7	1.42	0.2	1	0.8
5	1.04	0.8	1.22	0.35	1	0.9
7	1.02	0.8	1.17	0.33	1	0.9

$q_1 = 7500 \text{ nm}^2/\text{sec}^3$ ,  $q_2 = 25 \text{ nm}/\text{sec}$ ,  $q_3 = 0.1$ , and  $10^3$  MC samples were used per object. To enable comparisons with manual tracking, five independent, expert observers also tracked the 2D synthetic image sequences, using the freely available software tool MTrackJ [94].

#### 3.5.1.4 Tracking Results

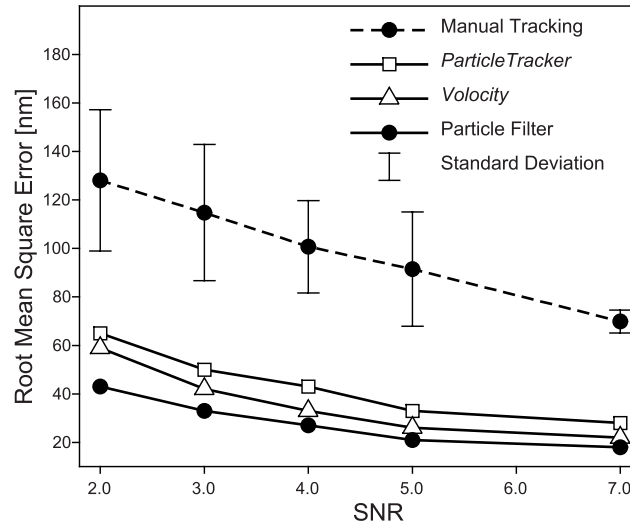
First, using the 2D synthetic image sequences, we compared the ability of our algorithm, *Volocity*, and *ParticleTracker* to track objects correctly, despite possible

object appearances, disappearances, and interactions or crossings. The results of this comparison are presented in Table 3.1. Two performance measures are listed:  $r_0$ , which is the ratio between the number of tracks produced by the algorithm and the true number of tracks present in the data ( $N_{tr}$ ), and  $r_1$ , which is the ratio between the number of correctly detected tracks and the true number of tracks. Ideally, the values for both ratios should be equal to 1. A value of  $r_0 > 1$  indicates that the method produced broken tracks. The main cause of this is the inability to resolve track intersections in some cases (see Fig. 3.4 for an example). In such situations the method either initiates new tracks after the object interaction event (because during the detection stage only one object was detected at that location, see Fig. 3.4), increasing the ratio  $r_0$ , or it incorrectly interchanges the tracks before and after the interaction (see Fig. 3.5 for an example), lowering the ratio  $r_1$ . From the results in Table 3.1 and the examples in Figs. 3.4 and 3.5, it clearly follows that our PF method is much more robust in dealing with object interactions. The scenario in the latter example causes no problems for the PF, as, contrary to two other methods, it exploits information about object appearance. During the measurement-to-track association, the PF favors measurements that are close to the predicted location and that have an elongation in the predicted direction of motion. In some cases (see Fig. 3.6 for an example), all three methods fail, which generally occurs when the interaction is too complicated to resolve even for expert biologists.

Using the same data sets and tracking results, we calculated the RMSE in object position estimation, as a function of SNR. To make a fair comparison, only the results of correctly detected tracks were included in these calculations. The results are shown in Fig. 3.7. The localization error of our algorithm is in the range of 10–50 nm, depending on the SNR, which is approximately 2–3 times smaller than for manual tracking. The error bars represent the interobserver variability for manual tracking, which, together with the average errors, indicate that the performance of manual tracking degrades significantly for low SNRs, as expected. The errors of the three automated methods show the same trend, with our method being consistently more accurate than the other two. This may be explained by the fact that, in addition to object localization by center-of-mass estimation, our hierarchical search performs further localization refinement during the second step (3.22). The RMSE in Fig. 3.7 is larger than in Fig. 3.3, because, even though only correct tracks were included, the accuracy of object localization during multiple object tracking is unfavorably influenced at places where object interaction occurs.

Our algorithm was also tested on the 3D synthetic image sequences as described, using 20 MC simulations. The RMSEs for the observation model  $L_{SG}$  ranged from  $\approx 30$  nm (SNR = 7) to  $\approx 70$  nm (SNR = 2). These errors were comparable to the errors produced by *Volocity* (in this test, *ParticleTracker* was excluded, as it is limited to tracking in 2D+t). Despite the fact that the axial resolution of the imaging system is approximately three times lower, the localization error was not affected dramatically relative to the 2D+t case. The reason for this is that in 3D+t data, we have a larger number of informative image elements (voxels). As a result, the difference in the RMSEs produced by the estimators employed in our algorithm and in *Volocity* is less compared to Fig. 3.7.





**Figure 3.7.** The RMSE in object position estimation as a function of SNR for our algorithm (Particle Filter) versus the two other automatic methods (*Volocity* and *ParticleTracker*) and manual tracking (five observers) based on synthetic image data.

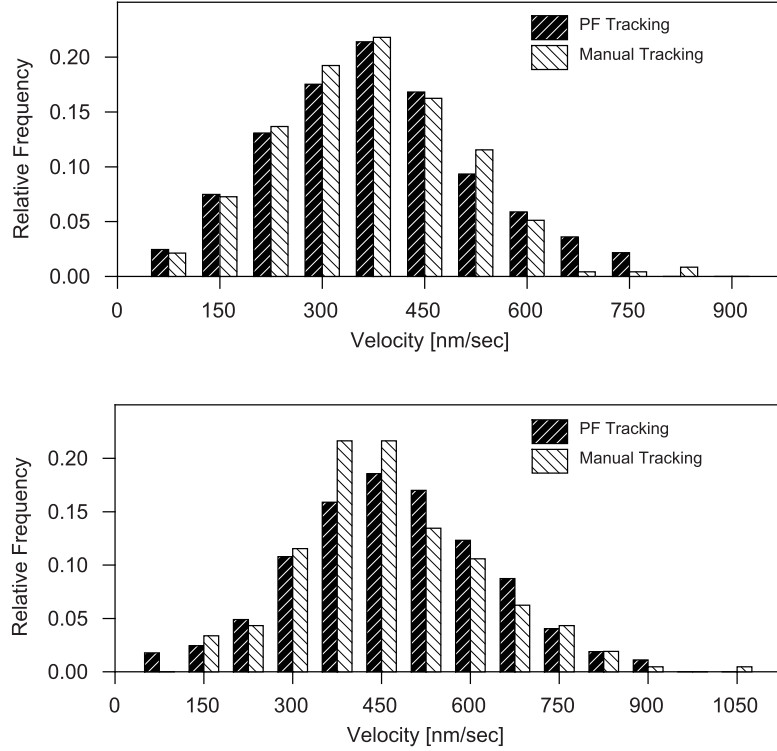
### 3.5.2 Evaluation on Real Data

#### 3.5.2.1 Image Acquisition

In addition to the computer generated image data, real 2D fluorescence microscopy image sequences of MT dynamics were acquired. COS-1 cells were cultured and transfected with GFP-tagged proteins as described [5,155]. Cells were analyzed at 37°C on a Zeiss 510 confocal laser scanning microscope (LSM-510). In most experiments the optical slice separation (in the  $z$ -dimension) was set to 1  $\mu\text{m}$ . Images of GFP+TIP movements in transfected cells were acquired every 1–3.5 seconds. For different imaging setups, the pixel size ranged from  $70 \times 70 \text{ nm}^2$  to  $110 \times 110 \text{ nm}^2$ . Image sequences of 30–50 frames were recorded and movies assembled using LSM-510 software. Six representative data sets (30 frames of size  $512 \times 512$  pixels), examples of which are shown in Fig. 3.1, were preselected from larger volumes by manually choosing the regions of interest. GFP+TIP dashes were tracked in different cell areas. Instantaneous velocities of dashes were calculated simply by dividing measured or tracked distances between frames by the temporal sampling interval.

#### 3.5.2.2 Comparison with Manual Tracking

Lacking ground truth for the real data, we evaluated the performance of our algorithm by visual comparison with manual tracking results. In this case, the latter were obtained from two expert cell biologists, each of which tracked 10 moving MTs of

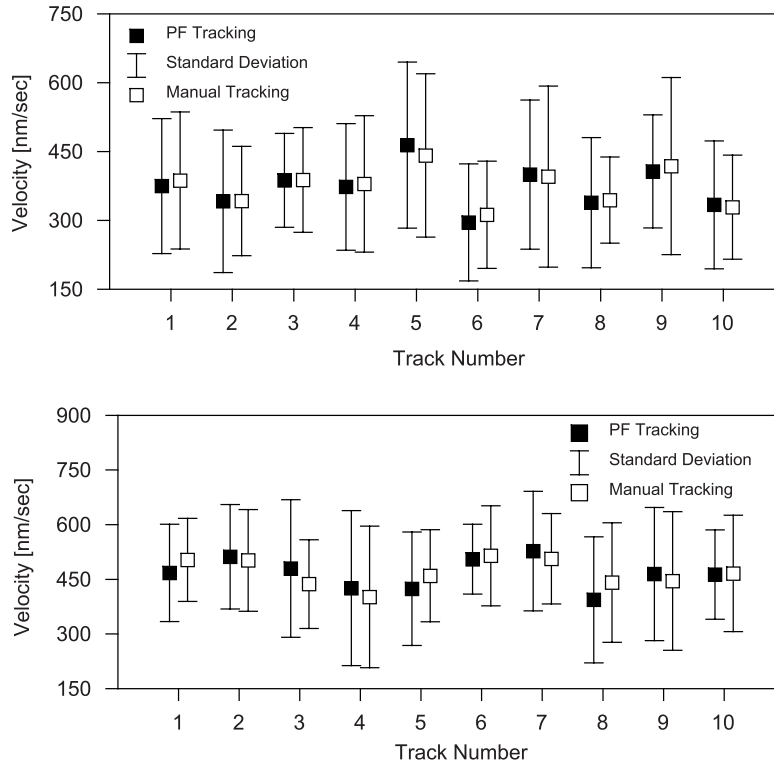


**Figure 3.8.** Examples of velocity distributions obtained with our automatic tracking algorithm versus manual tracking applied to real fluorescence microscopy image sequences of growing MTs. Results are shown for the data sets in Fig. 3.1(a) (top) and Fig. 3.1(f) (bottom).

interest by using the aforementioned software tool MTrackJ. The selection of target MTs to be tracked was made independently by the two observers. Also, the decision of which feature to track (the tip, the center, or the brightest point) was left to the observers. When done consistently, this does not influence velocity estimations, which is what we focused on in these experiments. The parameters of our algorithm (run with the model  $L_{SG}$ ) were fixed to the same values as in the case of the evaluation on synthetic data.

### 3.5.2.3 Tracking Results

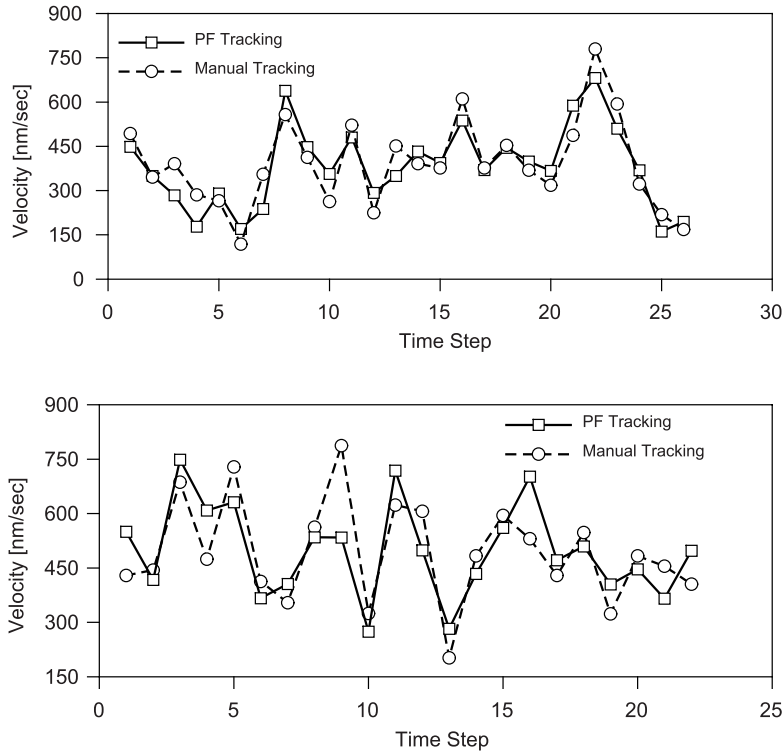
Distributions of instant velocities estimated using our algorithm versus manual tracking are presented in Fig. 3.8. The graphs show the results for the data sets of Fig. 3.1(a) and (f), for which  $\text{SNR} \approx 5$  and  $\text{SNR} \approx 2$ , respectively. A visual comparison of the estimated velocities per track, for each of the 10 tracks (the average track length was 13 time steps), is presented in Fig. 3.9, with more details for two



**Figure 3.9.** Results of velocity estimation for 10 representative MT objects in real fluorescence microscopy image sequences using our automatic tracking algorithm versus manual tracking for the data sets in Fig. 3.1(a) (top) and Fig. 3.1(f) (bottom). Shown are the mean values (black or white squares) and  $\pm 1$  standard deviation (bars) of the estimates.

representative tracks shown in Fig. 3.10. Application of a paired Student  $t$ -test per track revealed no statistically significant difference between the results of our algorithm and that of manual tracking, for both expert human observers ( $p \gg 0.05$  in all cases). Often, biologists are interested in average velocities over sets of tracks. In the described experiments, the difference in average velocity (per 10 tracks) between automatic and manual tracking was less than 1%, for both observers. Our velocity estimates are also comparable to those reported previously based on manual tracking in the same type of image data [155].

Finally, we present two different example visualizations of real data together with the results of tracking using our algorithm. Fig. 3.11 shows the results of tracking in the presence of photobleaching, which clearly illustrates the capability of our algorithm to initiate new tracks for appearing objects, to terminate tracks for disappearing objects, and to deal with closely passing objects. The rendering in Fig. 3.12 gives a

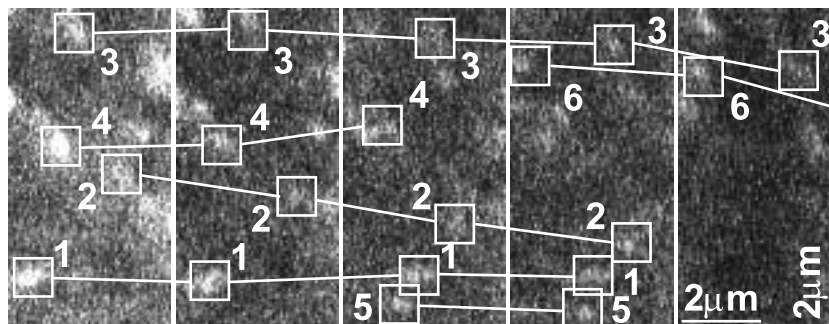


**Figure 3.10.** Velocity estimates per time step for our automatic tracking algorithm versus manual tracking. Results are shown for track numbers 4 (top) and 10 (bottom) in Fig. 3.9 (also from the top and bottom graphs, respectively).

visual impression of the full tracking results for a few time frames of one of the real data sets used in the experiments.

### 3.6 Discussion and Conclusions

In this chapter we have demonstrated the applicability of particle filtering for quantitative analysis of subcellular dynamics. Compared to existing approaches in this field, our approach is a substantial improvement for detection and tracking of large numbers of spots in image data with low SNR. Conventional methods, which perform object detection prior to the linking stage, use non-Bayesian maximum likelihood or least squares estimators. The variance of those estimators is larger than the variance of the MMSE estimator [11], for which some prior information about the estimated parameters is assumed to be known. In our case, this information is the prediction of the object position according to the motion model. This step, which optimally exploits available temporal information, makes our probabilistic tracking approach

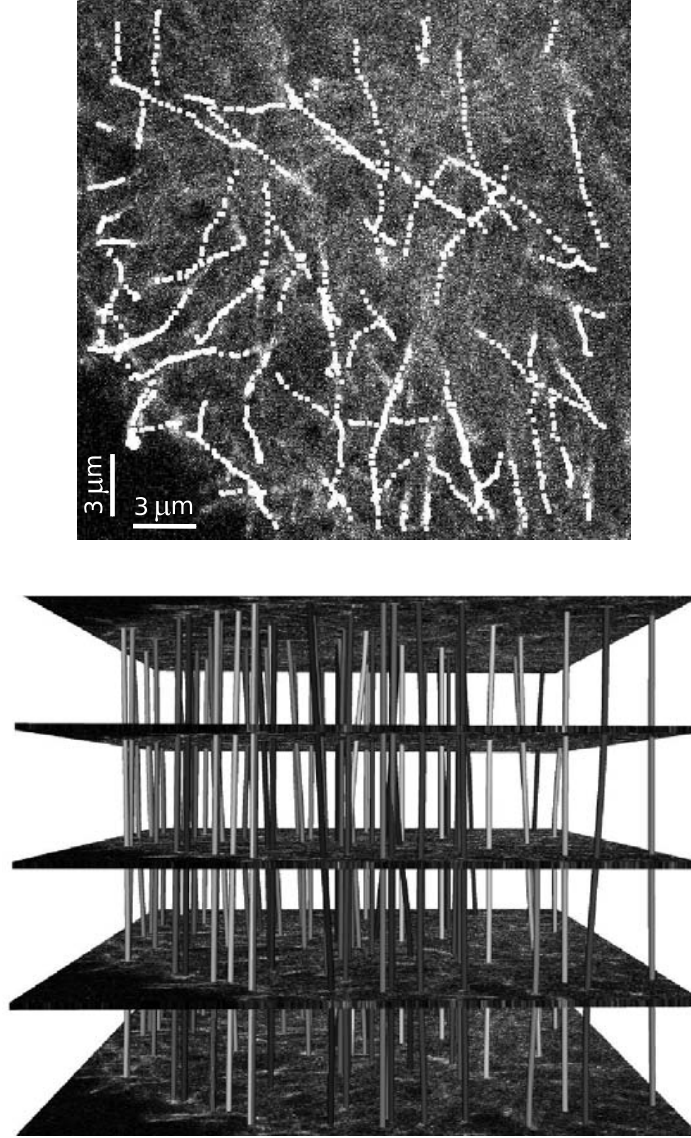


**Figure 3.11.** Results (six tracks) of automatically tracking MTs (bright spots) in the presence of photobleaching, illustrating the capability of our algorithm to capture newly appearing objects (tracks 5 and 6) and to detect object disappearance (for example track 4). It also shows the robustness of the algorithm in the case of closely passing objects (tracks 1 and 5).

perform superior in the presence of severe noise in comparison with existing frame-by-frame approaches, which break down at  $\text{SNR} < 4\text{--}5$  [26, 32]. As the experiments show, contrary to two other popular tracking tools, our algorithm still yields reliable tracking results even in data with SNR as low as 2 (which is not uncommon in practice). We note that the comparison with these two-stage tracking approaches mainly evaluated the linking parts of the algorithms, as the detection part is based on thresholding, and the parameters for that stage were optimized manually until all the desired objects were localized. In practice, since these algorithms were not designed specifically to deal with photobleaching effects, they can be expected to perform worse than reported here.

The results of the experiments on synthetic image data suggest that our algorithm is potentially more accurate than manual tracking by expert human observers. The experiments on real fluorescence microscopy image sequences from MT dynamics studies showed comparable performance. This is explained by the fact that in the latter experiments, we were limited to comparing distributions and averages (Figs. 3.8 and 3.9), which may conceal small local discrepancies, especially when the objects' velocities vary over time. Instant velocities were also analyzed per track (Fig. 3.10) but could not be quantitatively validated due to the lack of ground truth. Nevertheless, the results indicate that our algorithm may replace laborious manual procedures. Currently we are evaluating the method also for other biological applications to further demonstrate its advantages over current means of manual and automated tracking and quantification of subcellular dynamics. Our findings encourage use of the method to analyze complex biological image sequences not only for obtaining statistical estimates of average velocity and life span, but also for detailed analyses of complete life histories.

The 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 [121]), a public domain and platform independent image pro-

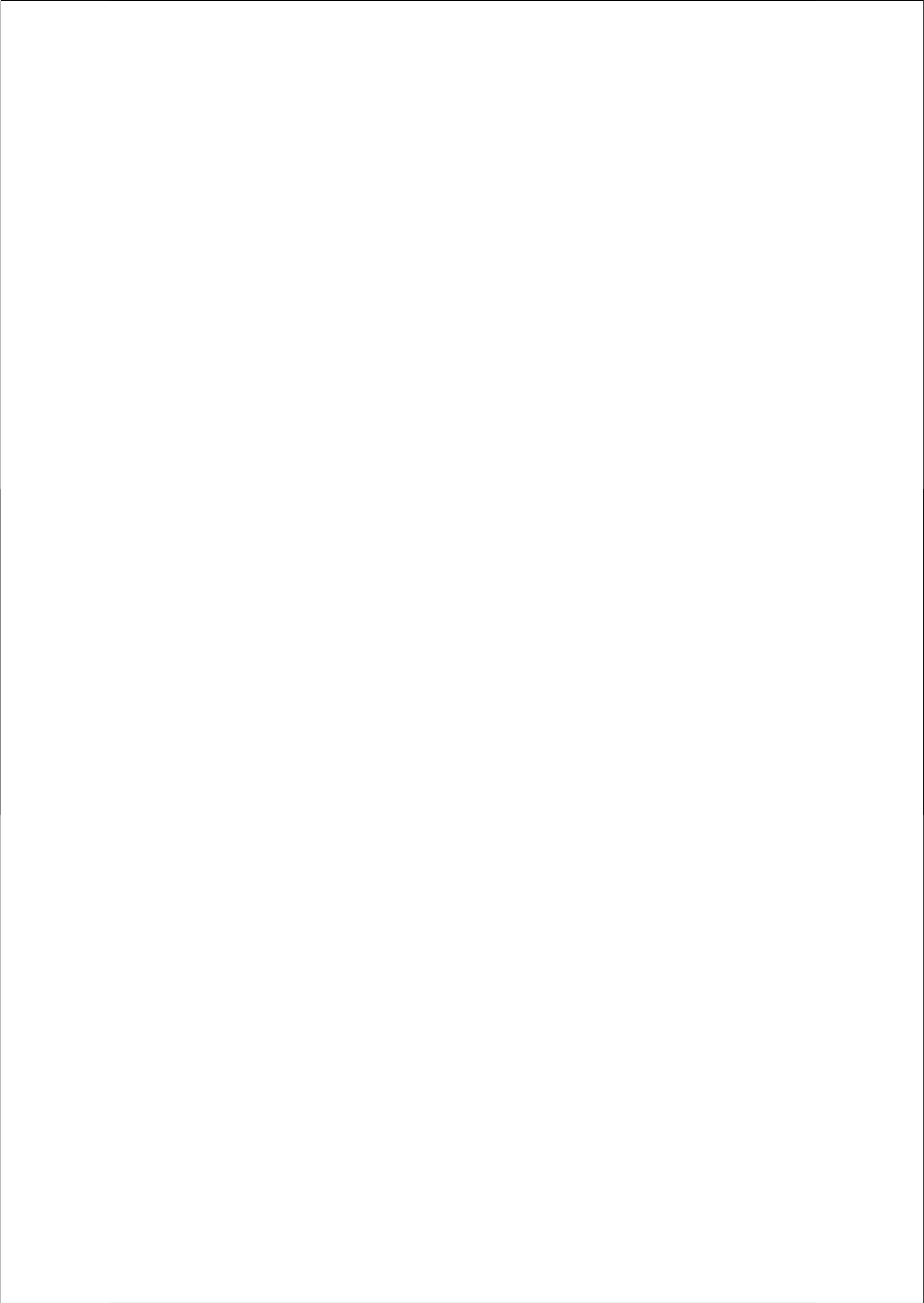


**Figure 3.12.** Visualization of tracking results (80 tracks) produced by our algorithm in the case of the real fluorescence microscopy image sequence of Fig. 3.1(a). Left: Trajectories projected on top of one of the frames, giving an impression of the MT dynamics in this image sequence. Right: Five frames from the sequence (time is increasing from bottom to top) with the trajectories rendered as small tubes connecting the frames. The rendering was accomplished using a script developed in-house based on the Visualization Toolkit [136].

cessing program used abundantly in biomedical image analysis [1]. Running on a regular PC (a Pentium IV with 3.2 GHz CPU and 3 GB of RAM) using the Java Virtual Machine version 1.5, the processing time per object per frame using  $10^3$  MC particles is about 0.3 sec. This cost is independent of image size, because all computations are done only for measurements falling inside the gates (defined for each track). We expect that faster execution times are still possible, after further optimization of the code. In the near future the algorithm will be integrated into a user-friendly software tool which will be made publically available.

The recursive nature of the proposed method (only the measurements up to time  $t$  are required in order to estimate the object positions at time  $t$ ) can be effectively utilized to dramatically increase the throughput of live cell imaging experiments. Usually time-lapse imaging requires constant adjustment of the imaging field and focus position to keep the cell of interest centered in the imaged volume. There are basically two methods to track moving objects with a microscope. Most commonly, images are acquired at a fixed stage and focus position and the movements are analyzed afterwards, using batch image processing algorithms. The second possibility, rarely implemented, is to program the microscope to follow the movements of the cell automatically and keep it in the field of view. Such tracking systems have been developed previously [82, 119, 120], but they are either hardware-based or not easily portable to other microscopes. Using the proposed software-based tracking method, however, it can be implemented on any fluorescence microscope with motorized stage and focus. The prediction step of the algorithm can be used to adapt the field of view and steer the laser in the direction of moving objects. This also suggests a mechanism for limiting laser excitation and thereby reducing photobleaching.





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

---

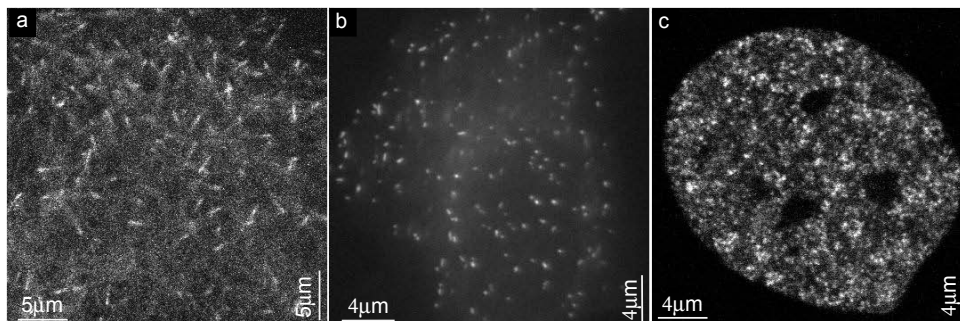
*Essentially, all models are wrong, but some are useful.*

— 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  $\approx 5$ –6 (b) to  $\approx 2$ –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]

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

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

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

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

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

where  $\mathbf{R} = \mathbf{R}(\phi_t)$  is a rotation matrix,  $\mathbf{\Sigma}_t = \text{diag}[\sigma_{\max,t}^2, \sigma_{\min,t}^2]$ ,  $\mathbf{m}_t = (i\Delta_x - x_t, j\Delta_y - y_t)^T$ , and  $\tan \phi_t = \dot{y}_t/\dot{x}_t$ . Here, each pixel  $(i, j)$  is assumed to correspond to a rectangular area of dimensions  $\Delta_x \times \Delta_y$  nm<sup>2</sup>. 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  $G_\sigma$  is the Gaussian kernel with scale  $\sigma$  (in practice we use  $\sigma = 30\text{--}50$  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  $G_\sigma$ , 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  $\pi_d$  of the likelihood (defined by a  $\chi^2$ -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

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

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

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  $\chi_L^2$  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  $\pi_d$ , 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  $\pi_d$  is defined as  $\hat{\pi} + 3\sigma_{\pi,t}$  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):  $\sigma_{\max}$  and  $\sigma_{\min}$  (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  $\Pi$  (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  $\gamma$  (importance sampling mixture ratio, Section 4.4.2). The remaining parameters are precomputed automatically from the image data:  $\alpha$  (photobleaching rate, Section 4.2.3),  $\pi_d$  (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  $\gamma$ .
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  $\pi_d$  (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  $\alpha/\beta$ -tubulin heterodimers and play the role of conveyor 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):

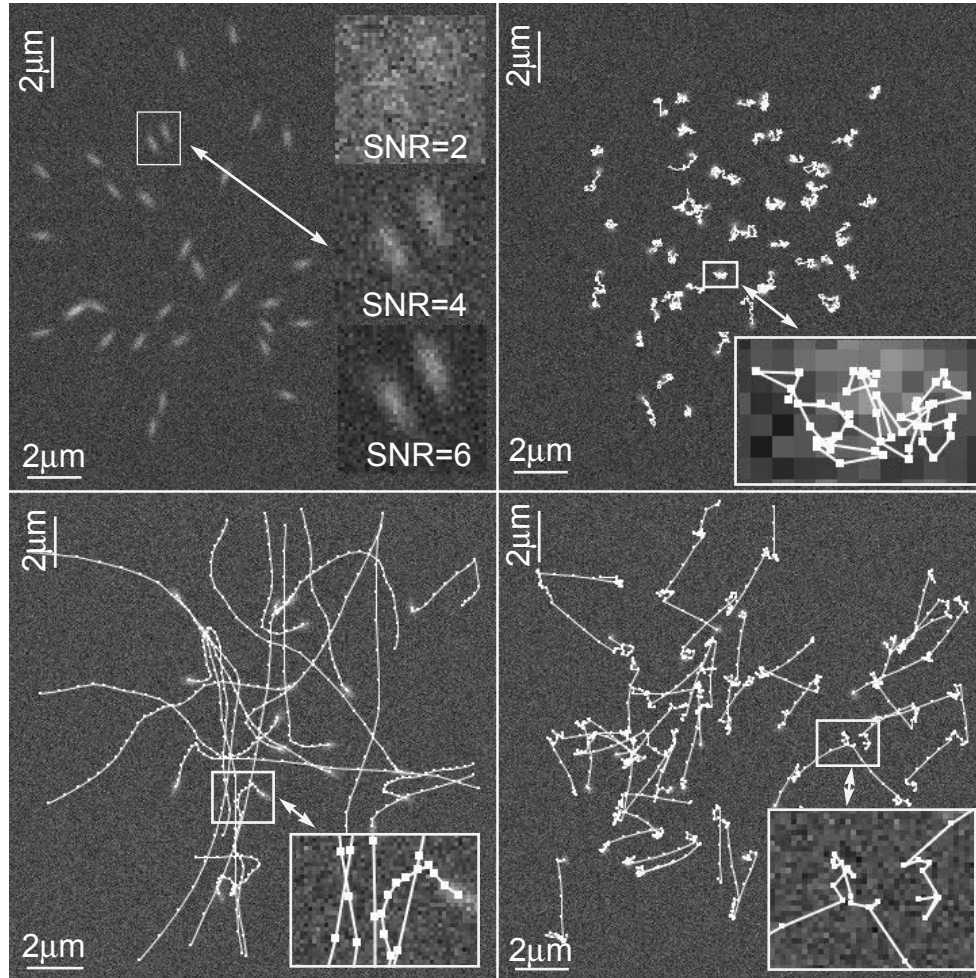
$$\Pi = \begin{pmatrix} 0.9 & 0.1 \\ 0.2 & 0.8 \end{pmatrix}. \quad (4.47)$$

Realistic 2D image sequences (Fig. 4.2), consisting of 30–100 frames ( $T = 1$  sec.) of  $512 \times 512$  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  $q_{1,1} = q_{1,2} = 5000$  nm<sup>2</sup>/sec<sup>2</sup>,  $q_2 = 0.04$ , and  $q_3 = 100$  nm<sup>2</sup>.

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  $\chi_2$ -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.

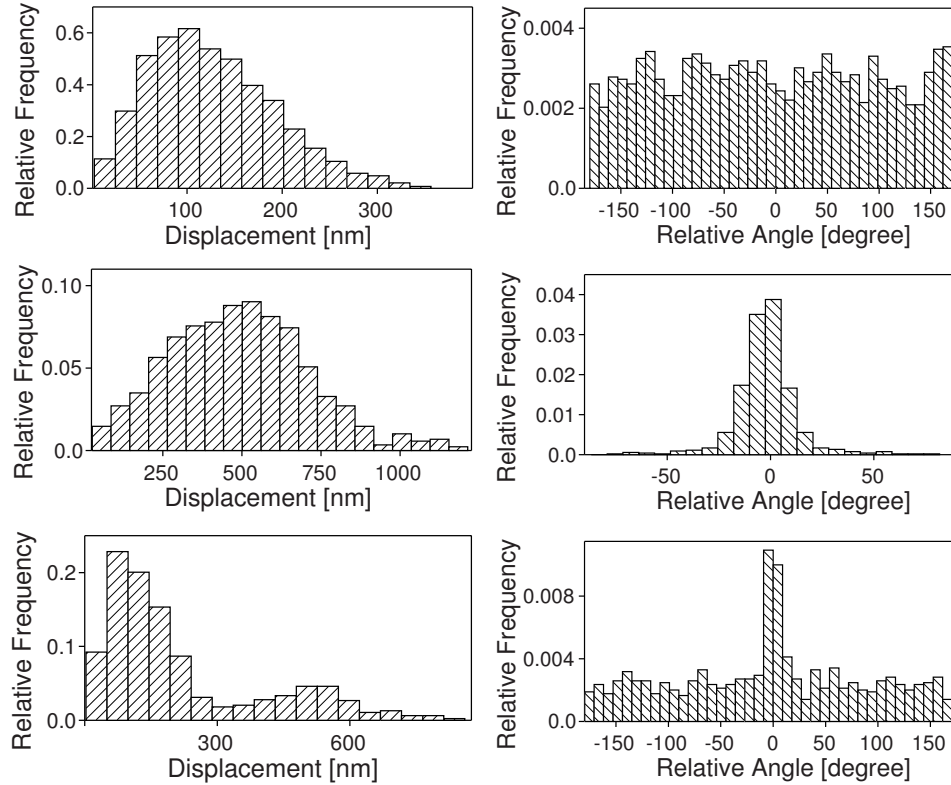
For all three applications, the estimated parameters of interest (average velocity, run length, drift coefficient, et cetera) are different, but all of them can be derived



**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





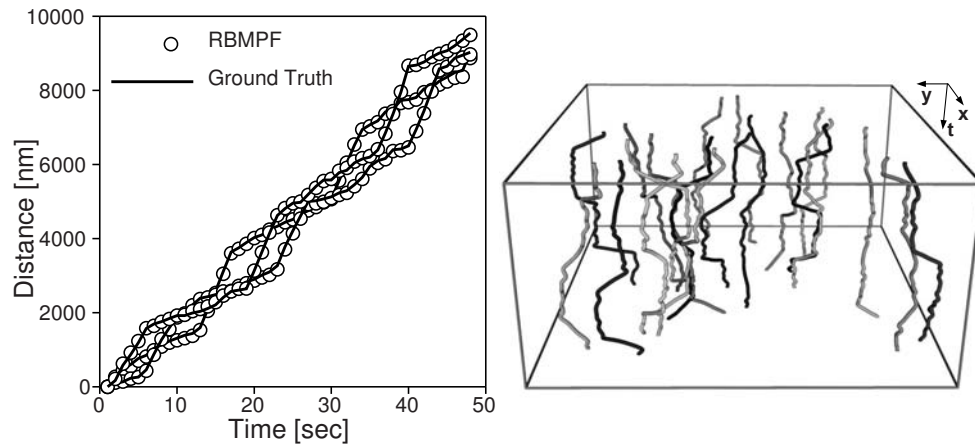
**Figure 4.3.** Distribution of the displacements and relative angles for the modeled synthetic data according to the described dynamics model for ARs (top row), MTs (middle row), and vesicles (bottom row).

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
			PF	RBMPF	PF	RBMPF	RBMPF
	RMSE [nm]	StDev [nm]	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



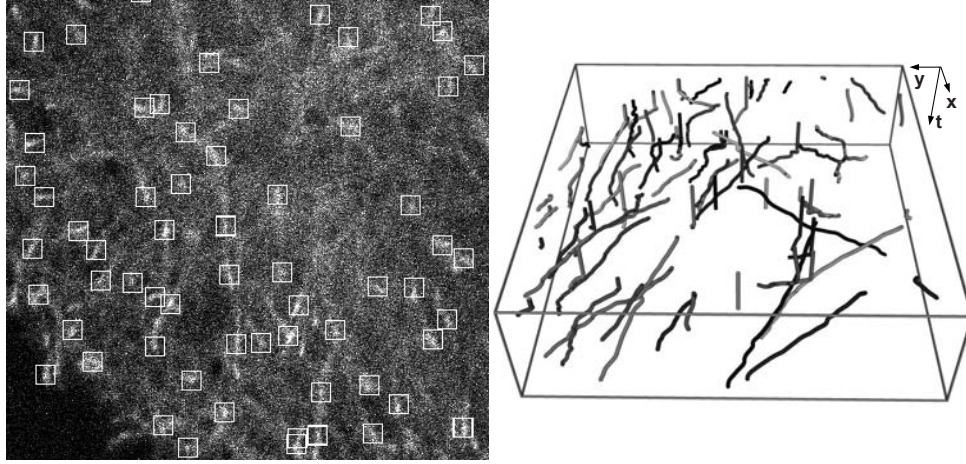
**Figure 4.4.** Results (three tracks) of automatically tracking vesicles that exhibit switching between fast and slow motion (left). The plot shows traveled distance versus time. Steep slopes represent short stages of fast motion, with almost constant velocity, which are preceded or followed by longer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

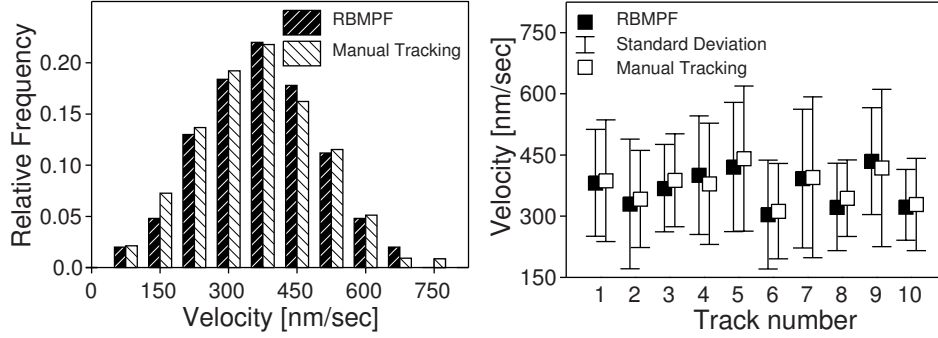
#### 4.6.3.1 Microtubules

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  $512 \times 512$  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

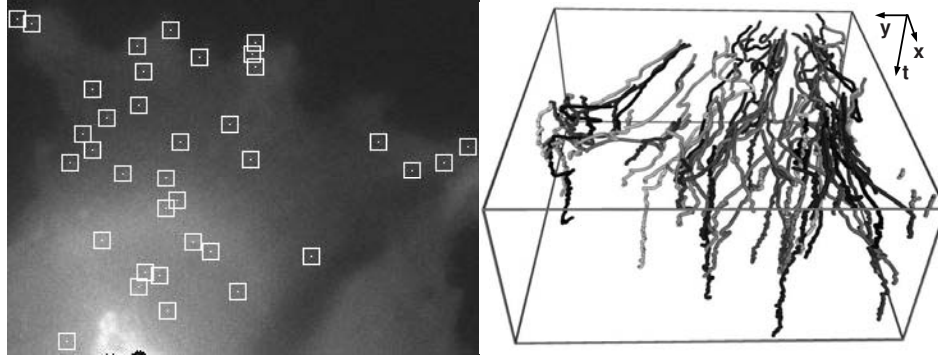
For peroxisome and Rab6 imaging [58], we used HeLa cells and PEX3-GFP fusion, which was a gift from B. Distel [181]. The HeLa cell line is the oldest cell line



**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  $\pm 1$  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  $1344 \times 1024$  pixels of size  $64 \times 64$  nm<sup>2</sup>. 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  $\Pi$ . 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.

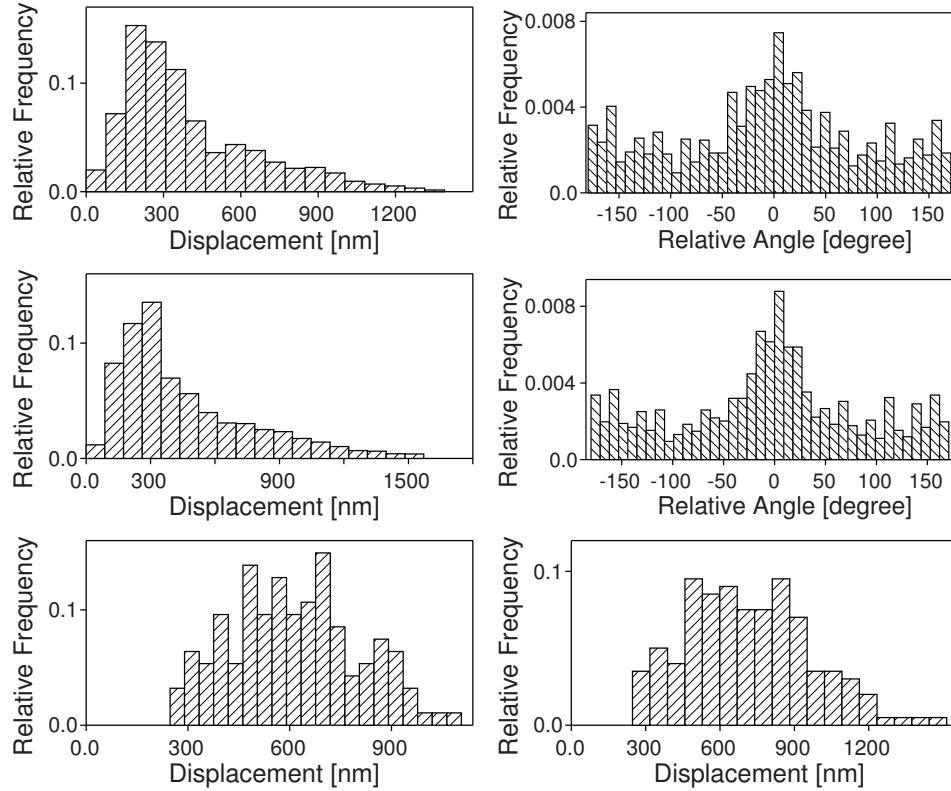
**Table 4.2.** Comparison of estimates of velocity [ $\mu\text{m}/\text{sec}$ ], the average ratio of fast motion time to total track time, and the probability of switching between the two motion stages of vesicles in HeLa-R and HeLa-L cells, based on manual tracking versus automatic tracking using the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

Estimate	Manual Tracking		Automatic Tracking	
	HeLa-R	HeLa-L	HeLa-R	HeLa-L
Velocity	$0.72 \pm 0.23$	$0.83 \pm 0.35$	$0.75 \pm 0.28$	$0.80 \pm 0.31$
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

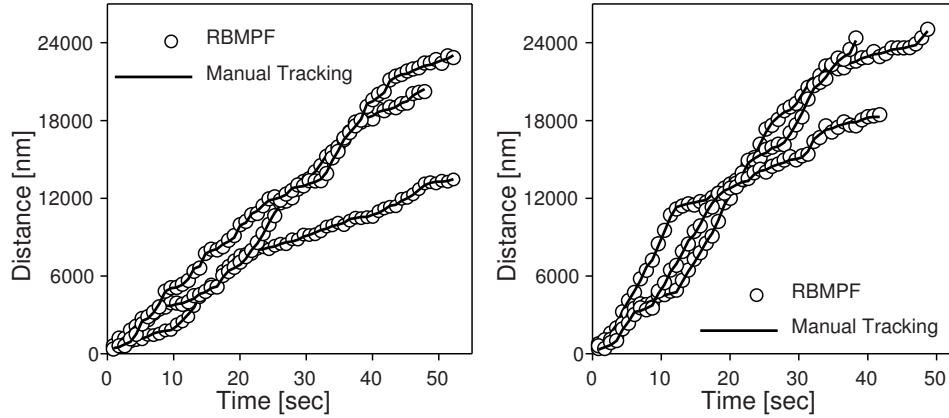
For AR imaging [170], YFP emission was detected in Hep3B cells expressing YFP and CFP (yellow and cyan variants of GFP respectively) double-tagged AR. An LSM-510 microscope was used to acquire images at a rate of 1 frame per 12.6 seconds. The resulting image sequences consisted of 80–180 frames of  $512 \times 512$  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-

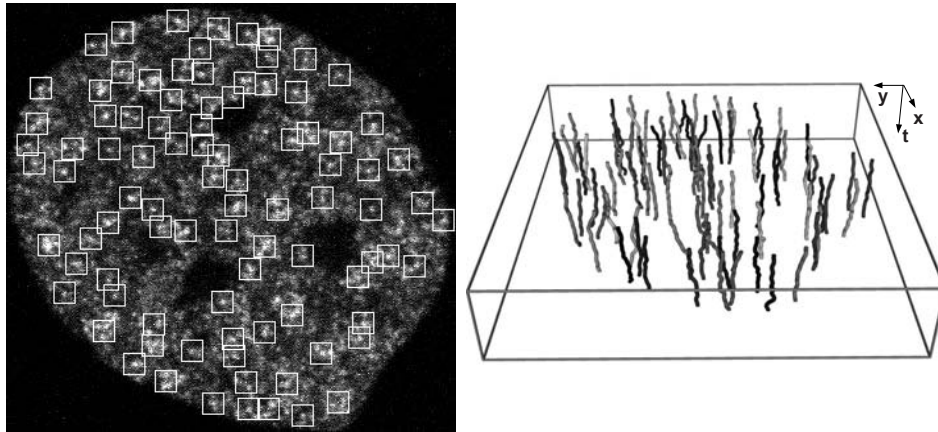


**Figure 4.8.** Distribution of the displacements and relative angles for 30 automatically obtained vesicle tracks for HeLa-R (top row) and HeLa-L (middle row) cells, and the filtered (thresholded) displacement histograms (bottom row), used for velocity analysis during fast motion stages for HeLa-R (left)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



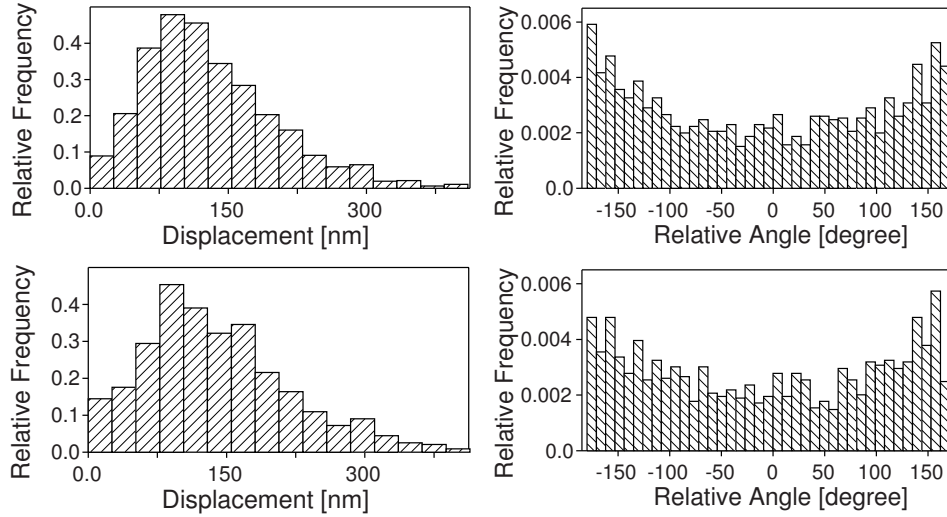
**Figure 4.9.** Results (three tracks) of automatically tracking vesicles that exhibit switching between fast and slow motion for HeLa-R (left) and HeLa-L (right) cells. The plots show traveled distance versus time. Steep slopes represent short stages of fast motion, with almost constant velocity, which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



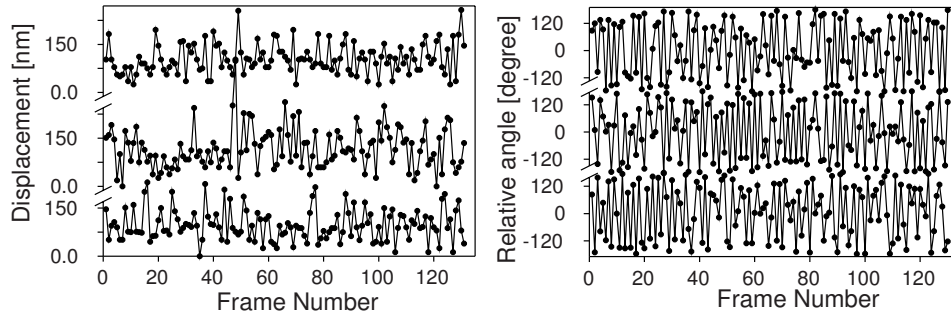
**Figure 4.10.** Single frame (left) from a time-lapse image data set acquired for studying AR dynamics, with the objects detected by our algorithm marked with white squares, and the results (right) of tracking using the algorithm. All objects with sufficiently high local SNR as determined by the  $h$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



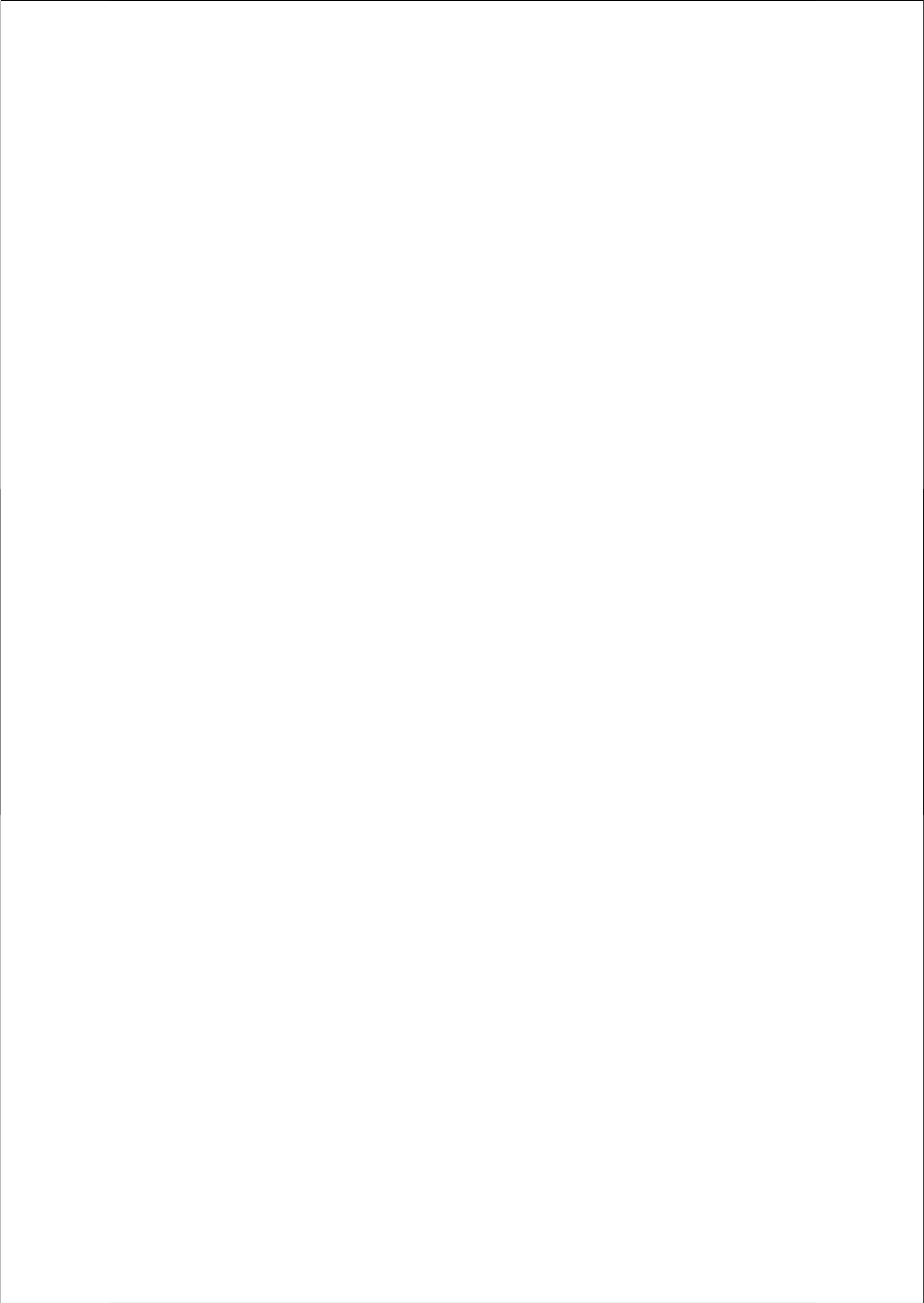
**Figure 4.12.** Examples of the velocity (left) and relative angle (right) time series for three automatically analyzed AR tracks.

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.





# Microtubule Dynamics Analysis using Kymographs and Variable-Rate Particle Filters

---

*It is a truth very certain that when it is not in our power to  
determine what is true we ought to follow what is most probable.*

— RENÉ DESCARTES,  
*Discours de la Méthode* (1637)

**Abstract** — Studying the dynamics of intracellular objects is of fundamental importance in understanding healthy life at the molecular level and to develop drugs to target disease processes. One of the key technologies to enable this research is the automated tracking and motion analysis of these objects in microscopy image sequences. To make better use of the spatiotemporal information than common frame-by-frame tracking methods, two alternative approaches have recently been proposed, based on either Bayesian estimation or space-time segmentation. In this chapter, we propose to combine the power of both approaches, and develop a new probabilistic method to segment the traces of the moving objects in kymograph representations of the image data. It is based on variable-rate particle filtering and uses multiscale trend analysis of the extracted traces to estimate the relevant kinematic parameters. Experiments on realistic synthetically generated images as well as on real biological image data demonstrate the improved potential of the new method for the analysis of microtubule dynamics *in vitro*.

---

Based upon: I. Smal, I. Grigoriev, A. Akhmanova, W. J. Niessen, E. Meijering, “Microtubule Dynamics Analysis using Kymographs and Variable-Rate Particle Filters”, *submitted*.

## 5.1 Introduction

**M**otion analysis of subcellular objects plays a major role in understanding fundamental dynamical processes occurring in biological cells. Since many diseases originate from a disturbance or failure of one or more of these processes, their study is of interest not only to life scientists, but also to pharmaceutical companies in their attempts to develop adequate drugs. Even though many intracellular interaction mechanisms are well understood these days, many questions still remain unanswered. In some cases, where the analysis in living cells (in cultures or *in vivo*) is confounded by other intracellular processes, it makes sense to study the objects of interest *in vitro*, where the influence of other structures or processes is removed, reduced, or known [14, 101].

Intracellular dynamics is usually visualized using advanced microscopy imaging techniques, such as fluorescence confocal microscopy, where the objects of interest are labeled with fluorescent proteins. Alternatively, differential interference contrast (DIC) microscopy can sometimes be used, which does not require labeling [112, 156]. In either case, the optical resolution of the microscope is much lower (on the order of 100 nm) than the size of the objects of interest (on the order of nanometers), causing the latter to be imaged as blurred spots (without sharp boundaries) due to diffraction. The quality of the images is further reduced by high levels of measurement noise [112, 185]. Both types of distortions contribute to the ambiguity of the data, making automated quantitative image analysis an extremely difficult task.

In time-lapse microscopy, where hundreds to thousands of 2D or 3D images are acquired sequentially in time, the main task is to track the objects of interest (proteins, vesicles, microtubules, etc.) and compute relevant motion parameters from the extracted trajectories. In practice, manual tracking is labor intensive and poorly reproducible, and only a small fraction of the data can be analyzed this way. The vast majority of automatic tracking methods [52, 71, 95, 96, 132, 160, 161] developed in this field consist of two stages: 1) *detection* of objects of interest (independently in each frame), and 2) *linking* of detected objects from frame to frame (solving the correspondence problem). Since the methods employed for the first stage operate on data with low signal-to-noise ratio (SNR), the linking procedure in the second stage is faced with either many false positives (noise classified as objects) or false negatives (mis-detection of actually present objects).

Contrary to these two-stage tracking methods, which typically use only very few neighboring frames to address the correspondence problem, methods that make better use of the available temporal information usually show better results. Such trackers are either built within a Bayesian estimation framework [141, 142], which in any frame uses all available temporal information up to that frame, or they consider the 2D+t or 3D+t image data as one spatiotemporal 3D or 4D image, respectively, and translate the estimation of trajectories into a segmentation of spatiotemporal structures [17, 128].

In this chapter, we propose to combine the power of the latter two approaches, and develop a variable-rate particle filtering method that implements the Bayesian estimation framework for tracing spatiotemporal structures formed by transforming the original time-lapse microscopy image data into a special type of spatiotemporal

representation: kymographs [12, 22, 64, 130]. This combined approach, which to the best of our knowledge has not been explored before, results in more accurate extraction of the spatiotemporal structures (edge-like image structures in our case) compared to particle filtering applied directly to the image sequences on a per-frame basis.

The chapter is organized as follows. In Section 5.2, we describe the biological application considered in this chapter and the proposed methods to model, acquire, transform, preprocess, and analyze the image data. In Section 5.3, we present experimental results of applying our method to synthetic image sequences, for which ground truth was available, and to real DIC microscopy image data of microtubule dynamics. A concluding discussion of the main findings is given in Section 5.4.

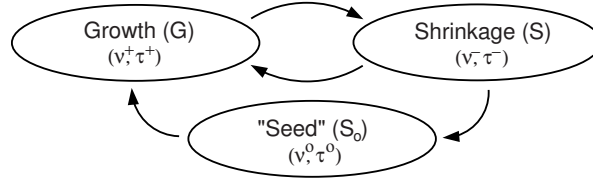
## 5.2 Methods

### 5.2.1 *In Vitro* Microtubule Dynamics Model

Microtubules (MTs) are polymers of tubulin, which assemble into hollow tubes (diameter  $\sim 25$  nm) in the presence of guanosine triphosphate (GTP), both *in vivo* and *in vitro* [37, 107]. *In vivo*, MTs are responsible for the support and shape of the cell and play a major role in several intracellular processes such as cell division, internal cell organization, and intracellular transport. MT dynamics (also referred to as dynamic instability) is highly regulated, both spatially and temporally, by a wide family of MT associated proteins (MAPs) [67]. To understand the specific interactions between regulatory factors and microtubules is of great interest to biologists. Misregulation of MT dynamics, for example, can lead to erroneous mitosis, which is a characteristic feature in neurodegenerative diseases.

Microtubule dynamic instability is a stochastic process of switching between growth and shrinkage stages, regulated by MAPs [99]. The growth velocity,  $\nu^+$ , depends on soluble tubulin concentration available for polymerization and GTP-tubulin association and dissociation rates. The shrinkage velocity,  $\nu^-$ , which is usually an order of magnitude higher than the growth velocity, is independent of tubulin concentration and is characterized only by the dissociation rate of guanosine diphosphate (GDP) tubulin from the depolymerizing end. The growth velocity *in vivo* can be up to 10 times faster than *in vitro*. Two other important events that characterize dynamic instability are *rescue* (switching from shrinkage to growth) and *catastrophe* (switching from growth to shrinkage) [99]. In practice, the analysis of MT dynamics includes estimation of  $\nu^+$ ,  $\nu^-$ , and the rescue and catastrophe frequencies,  $f_{\text{res}}$  and  $f_{\text{cat}}$ . The rescue rate *in vitro* is very low unless specific rescue factors are added to the assay and might be difficult to estimate reliably [101].

Recent studies reveal a special class of MAPs, plus-end-tracking proteins (+TIPs), that are able to accumulate at MT growing ends [6, 27, 67, 137]. The mechanisms by which +TIPs recognize MT ends have attracted much attention and several explanations have been proposed [4, 27, 67]. One way to understand the mechanism employed by individual +TIPs and the molecular mechanisms underlying their functions is by measuring the distribution and displacement of +TIPs in time. However, due to lack of robust and accurate automatic methods, the manual analysis usually is a labor



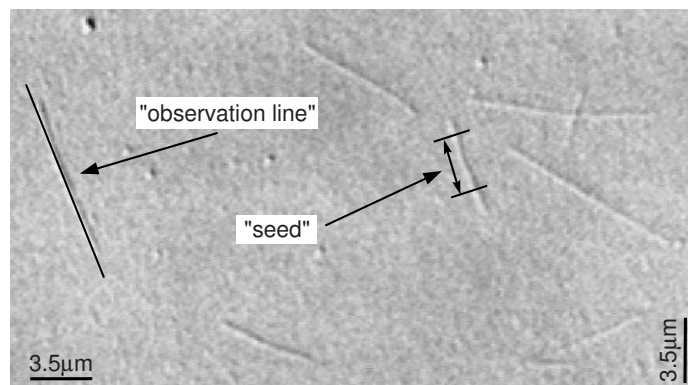
**Figure 5.1.** Dynamics model describing microtubule behavior *in vitro*.

intensive procedure which very likely leads to user bias and loss of important information. In the case of experiments in living cells it is extremely hard to decouple the effect of other regulators while studying +TIPs influence on MT dynamics. The advantage of *in vitro* investigation is the minimal environment in which the influence of various +TIPs can be dissected individually. Recent *in vitro* studies start to reveal the mechanisms of +TIPs end-tracking and the regulation of MT dynamics by individual +TIPs [182]. This can potentially lead to combining multiple +TIPs in order to reconstitute the *in vivo* MT dynamics and observe the collective effect of +TIPs.

The stochastic behavior of the MT tip can be modeled using a dynamical system with three states (Fig. 5.1):  $G$  (growth),  $S$  (shrinkage), and  $S_0$  (no dynamic activity). Each state is characterized by a velocity parameter  $\tilde{v} \in \{\nu^+, \nu^-, \nu^0\}$  and a duration time interval  $\tilde{\tau} \in \{\tau^+, \tau^-, \tau^0\}$ , describing the duration of the corresponding stage. The following state transitions are allowed:  $S_0 \rightarrow G$  (the MT starts to grow),  $G \rightarrow S$  (catastrophe),  $S \rightarrow G$  (rescue), and  $S \rightarrow S_0$  (the MT is completely disassembled). At each time point the MT can “stay” only in one of the states and for a period of time no longer than the corresponding  $\tilde{\tau}$  for that state. In our simulations, the time and velocity parameters are generated randomly (Section 5.3.1), and because of that it is allowed to “leave” the state  $S$  sooner than  $\tau^-$  if the MT is completely disassembled in shorter time. If after time  $\tau^-$  the MT was not disassembled completely (did not reach state  $S_0$ ), a rescue occurs ( $S \rightarrow G$ ) and the MT switches to growing. A similar three-stage model of MT dynamics can be designed for the *in vivo* situation. In this case, state  $S_0$  should be replaced with a state that corresponds to a “pause” event [37], and all the transitions (arrows in Fig. 5.1) should be bidirectional.

### 5.2.2 Imaging Technique and Kymographs

In our studies, the dynamic behavior of MTs is imaged using DIC microscopy [102], which is effectively used for biological specimens that cannot be visualized with sufficient contrast using bright-field microscopy. The resulting images (see Fig. 5.2 for an example) are similar to those obtained with phase-contrast microscopy and depict objects as black/white shadows on a gray background with good resolution and clarity. DIC microscopy works by separating a polarized light source into two beams that take slightly different paths through the sample and then converting changes in optical path length to a visible change in brightness [102]. The advantages of DIC over fluorescence microscopy is that the samples do not have to be stained. The main limitation of this imaging technique is its requirement for a thin and transparent sample of fairly similar refractive index to its surroundings.

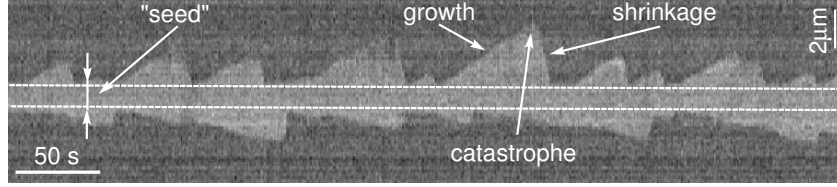


**Figure 5.2.** Example of a DIC microscopy image. Microtubule nucleation initiates from stable tubulin “seeds”. In the experiments, “observation lines” are drawn along MT bodies to construct kymographs.

Automatic analysis of MT behavior *in vitro* using time-lapse DIC microscopy images is a complicated task. The goal is to follow (track) the fast-growing (so called “plus”) end of each MT so as to obtain 2D paths in the image plane, from which all the parameters of interest (velocity and frequency estimates) can be computed. One of the main problems is that in DIC microscopy images, the object appearance (and especially the MT tip) depends on the imaging conditions (the relative angle between the sample and the microscope polarization prism) and cannot be easily modeled by appearance models, as in the case of fluorescence microscopy imaging. Additionally, the real object location is further obscured by diffraction, modeled by the point-spread function (PSF) of the microscope.

Another issue that requires careful consideration is the temporal sampling rate. In our experiments, images are acquired every second, which is in fact a quite high sampling rate taking into account how slowly microtubules grow *in vitro* (30-40 nm per second). This relatively high sampling rate is both a blessing and a curse. It is a blessing because it allows one to observe the motion in more detail and possibly detect rare and extraordinary movements that would otherwise go unnoticed. It is also a curse, however, as the growth and shrinkage velocities are usually such that the change in MT length from one frame to the next is (much) less than one pixel (in our experiments, the pixel size is  $80 \times 80 \text{ nm}^2$ ), even if the *spatial* sampling is done at the Nyquist rate. This is on the same order as the positional estimation errors made by manual or automatic approaches [141]. As a result, instant velocity estimates ( $\nu^+$  or  $\nu^-$ ) computed as the ratio of positional change over elapsed time between two consecutive frames, are doomed to be highly inaccurate.

In order to exploit all image data and at the same time obtain more accurate results, we abandon the idea of frame-by-frame tracking of objects directly in the original data, and we propose to base the estimation of motion parameters on a transformation of the data that is more amenable to multiscale analysis. Specifically, we propose to use a kymograph representation [130] (also called a kymoimage in this



**Figure 5.3.** Example of a kymograph obtained from the DIC microscopy images, showing the dynamics of both microtubule ends.

chapter) for each MT. It is constructed by defining (manually or automatically) an “observation line”  $L$  (Fig. 5.2) in the original image along the MT body. The length of  $L$  should approximately equal the maximum expected length of MTs in the sample. Image intensity values are then sampled equidistantly along  $L$ , yielding a vector of “measurements” at time  $t$ ,  $J_t = \{J_t(j) : j = 1, \dots, Y\}$ , where  $Y$  is the number of samples for the selected MT in every image frame. In practice, to increase the SNR, the measurements  $J_t(j)$  are obtained by averaging pixel values in the vicinity of  $j$ , along a line perpendicular to  $L$ . The resulting kymoimage (see Fig. 5.3 for an example),  $I(t, y) = \{J_t : t = 1, \dots, T\}$ , is the collection of measurement vectors, where every column  $t$  contains the measurements  $J_t$  as pixel values, and  $T$  is the number of frames in the image sequence. In our experiments, MT nucleation from stable tubulin oligomers was studied [28]. These “seeds” always remain present and cannot be completely disassembled. In the kymoimages (Fig. 5.3) they are clearly visible as a bright horizontal strip.

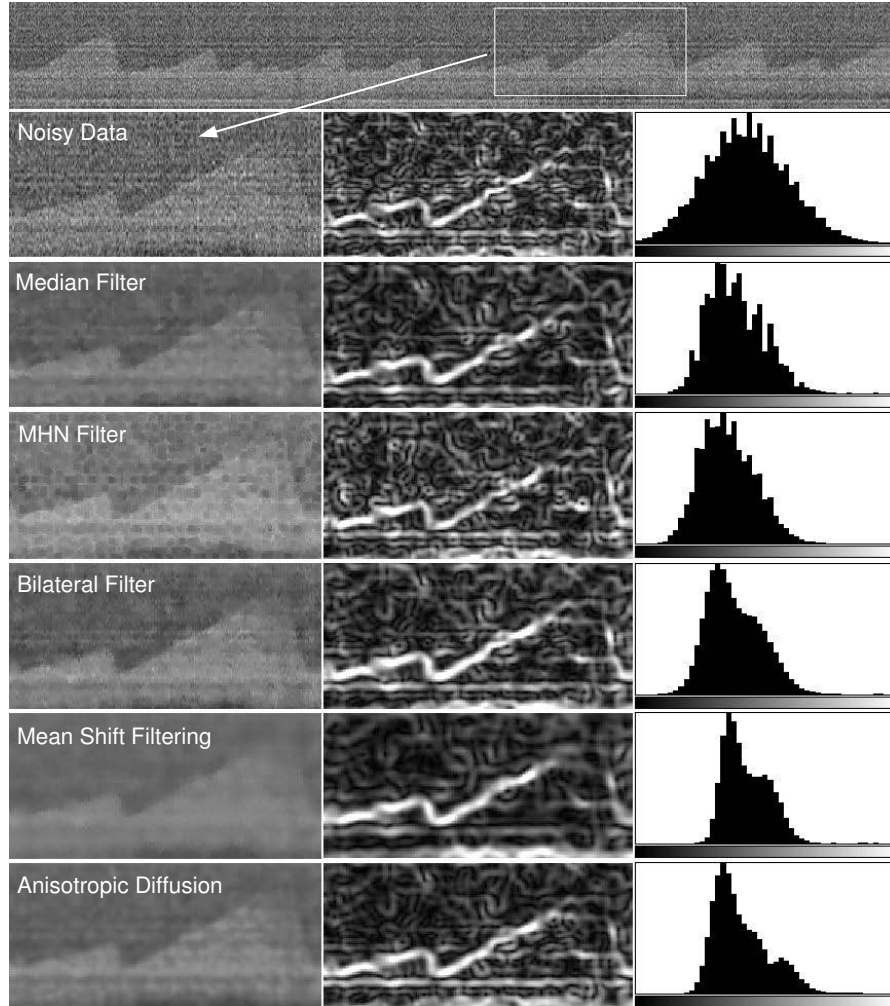
To estimate the kinematic parameters of interest from the kymoimages, the edge location  $y(t)$  (corresponding to the MT tip) should be accurately extracted (slopes should be preserved). In kymoimages, the instant velocity  $\nu$  at any time  $t'$  is estimated as  $\nu = (dy/dt)_{t=t'} = \tan(\varphi)$ , where  $\varphi$  is the angle between the time axis and the tangent to  $y(t)$  at  $t = t'$ . As a result, small errors in the angle estimates may lead to large errors in the velocity estimation, due to the nonlinearity introduced by the tangent (the closer  $\varphi$  is to 90 degrees, the larger the errors).

In this chapter, the analysis is conducted in three subsequent steps: 1) preprocessing, 2) edge extraction, and 3) multiscale trend analysis. Step 1 enhances the quality of the image using edge preserving filtering. Step 2 traces the edges by a particle filter capable of using multiscale measurements. Finally, step 3 analyzes the extracted edges by splitting them into relevant parts and performing linear approximation in order to compute all the necessary parameters. The three steps are described in more detail in the following subsections.

### 5.2.3 Edge Preserving Smoothing

The main challenge in estimating the growth velocity  $\nu^+$ , shrinkage velocity  $\nu^-$ , and the two transition frequencies  $f_{\text{res}}$  and  $f_{\text{cat}}$ , is to accurately segment the edges in the kymoimages (Fig. 5.3). Two main approaches to edge detection are differentiation and model fitting. In practice, differentiation, being a noise enhancing operation, requires some form of smoothing, which in turn entails the risk of blurring edge in-





**Figure 5.4.** Application of various edge preserving smoothing methods to our image data (top). The left column shows the results of smoothing, the middle column depicts the edge information extracted using the Gaussian derivatives, and the right column shows the distribution of intensity values in the smoothed images.

formation. Better results may be obtained by the use of nonlinear, edge preserving filters. Fig. 5.4 shows the results of applying the most frequently used nonlinear filtering techniques to our image data: the median filter [149], the maximum homogeneity neighbor (MHN) filter [49], the bilateral filter [163], the mean-shift filter [34], and anisotropic diffusion [116]. The examples clearly demonstrate that noise can be reduced to some extent while preserving edge information. However, they also show that edges may still not be clearly defined in (parts of) the image. Subsequent edge extraction by means of Gaussian differentiation [159] may result either in detection of

noisy background structures (at small scales), or in too much positional uncertainty (at larger scales), neither of which is acceptable for accurate slope estimation of the linear parts of the edge  $y(t)$ .

To overcome the problems caused by differentiation, we propose to use model fitting for edge detection, using particle filtering (PF) methods. The PF can be exploited to reduce the overload of fitting the model in every pixel position, by incorporating information about the edge model, the image noise distribution, and the probability of finding the edge in the neighborhood of a pixel, by taking into account the probability of edge existence at neighboring pixels. In this case, the use of edge preserving prefiltering is still advantageous. The PF mainly replaces the edge extraction part, which in differentiation based approaches such as Canny's algorithm [25] is usually based on hard thresholding.

#### 5.2.4 Variable-Rate Particle Filtering

The prefiltered kymoimage is an input for the next step, where particle filtering (PF) is performed to estimate the edge location  $y(t)$ . Particle filters [9, 126] implement the concept of Bayesian estimation, where at each time point  $t$  a system state  $\mathbf{x}_t$  is estimated on a basis of previous states, noisy measurements  $\mathbf{z}_t$  obtained from sensors, and prior knowledge about the underlying process [9]. For our application, the simplest working implementation of PF can be constructed with the state vector  $\mathbf{x}_t$ , which describes the position of the edge in every column  $t$  of the image  $I(t, y)$ , and the measurements  $\mathbf{z}_t$ , which are the intensity values in the corresponding column  $t$  of  $I(t, y)$ . Prior knowledge about the system is specified by the dynamics model, which describes the state transition process, and the observation model:

$$\mathbf{x}_t = f_t(\mathbf{x}_{t-1}, \mathbf{v}_t), \quad \mathbf{z}_t = g_t(\mathbf{x}_t, \mathbf{u}_t), \quad (5.1)$$

where  $f_t$  and  $g_t$  are possibly nonlinear functions and  $\mathbf{v}_t$  and  $\mathbf{u}_t$  are white noise sources. The choice of these functions is application specific and is given below. Alternatively, the same state estimation problem can be formulated by specifying two distributions,  $p(\mathbf{x}_t|\mathbf{x}_{t-1})$  and  $p(\mathbf{z}_t|\mathbf{x}_t)$ , instead of (5.1) [9, 126].

The solution of the state-space problem given by (5.1) is the posterior probability distribution function (pdf)  $p(\mathbf{x}_{0:t}|\mathbf{z}_{0:t})$ , where  $\mathbf{x}_{0:t} = \{\mathbf{x}_0, \dots, \mathbf{x}_t\}$  and  $\mathbf{z}_{0:t} = \{\mathbf{z}_0, \dots, \mathbf{z}_t\}$ , which can be found either exactly (when  $f_t$  and  $g_t$  are linear and  $\mathbf{v}_t$  and  $\mathbf{u}_t$  are Gaussian) using the Kalman filter [126] or, in the most general case, using approximations such as sequential Monte Carlo (MC) methods [9, 39]. In the latter case, the posterior pdf is approximated with a set of  $N_s$  MC samples (referred to as "particles"),  $\{\mathbf{x}_{0:t}^{(i)}, w_t^{(i)}\}_{i=1}^{N_s}$ , as

$$p(\mathbf{x}_{0:t}|\mathbf{z}_{0:t}) = \sum_{i=1}^{N_s} w_t^{(i)} \delta(\mathbf{x}_{0:t} - \mathbf{x}_{0:t}^{(i)}), \quad (5.2)$$

where  $\mathbf{x}_{0:t}^{(i)}$  describes one of the possible state sequences (path) and  $w_t^{(i)}$  is the weight indicating the probability of realization of that path. The solution using PF is given

by a recursive procedure that predicts the state from time  $t - 1$  to  $t$  and updates the weights based on newly arrived measurements  $\mathbf{z}_t$  as

$$\mathbf{x}_t^{(i)} \sim p(\mathbf{x}_t | \mathbf{x}_{t-1}^{(i)}) \text{ and } w_t^{(i)} \propto w_{t-1}^{(i)} p(\mathbf{z}_t | \mathbf{x}_t^{(i)}), \quad (5.3)$$

$i = 1, \dots, N_s$ . The minimum mean square error (MMSE) or maximum a posteriori (MAP) estimators of the state can be easily obtained from  $p(\mathbf{x}_{0:t} | \mathbf{z}_{0:t})$  [9].

Commonly, the state sampling rate is determined by the rate at which the measurements arrive. In the application under consideration, where the MT dynamics is characterized by prolonged periods of smoothness (growth and shrinkage stages) with infrequent sharp changes (rescue and catastrophe), it is possible to obtain a much more parsimonious representation of the MT tip trajectory if the state sampling rate is adapted to the nature of the data – more state points are allocated in the regions of rapid variation and relatively fewer state points to smoother sections. Unfortunately, this idea cannot be implemented using the standard PFs because the number of state points, which would typically be much smaller than the number of observations, is random and unknown beforehand. In order to deal with this randomness, variable-rate particle filtering (VRPF) methods have been proposed recently [56, 57]. The VRPF can be compared to the more conventional interactive multiple models (IMM) approach, which uses switching between a discrete set of candidate dynamical models [11, 52], but was shown to outperform IMM in most cases [56]. The VRPF, which was initially proposed for tracking of highly maneuvering targets [56], is nowadays successfully applied in other fields, for example DNA sequencing [61], but has not been investigated before in microscopy.

Contrary to the standard state-space approach, where the state variable  $\mathbf{x}_t$  evolves with time index  $t$ , within the VRPF framework the state  $\mathbf{x}_k$  is defined as  $\mathbf{x}_k = (\theta_k, \tau_k)$ , where  $k \in \mathbb{N}$  is a discrete state index,  $\tau_k \in \mathbb{R}^+ > \tau_{k-1}$  denotes the arrival time for the state  $k$ , and  $\theta_k$  denotes the vector of variables necessary to parametrize the object state. In tracking applications, the vector  $\theta_k$  includes variables such as position, velocity, heading, etc. For our application, we define  $\theta_k = (y_k, v_k)$ , where  $y_k$  is the edge position at time  $\tau_k$  along the observation line  $L$ , and  $v_k = (dy/dt)_{t=\tau_k}$  describes the direction of the edge at  $t = \tau_k$  in the image  $I(t, y)$ . Similarly to the standard PF, it is assumed that the state sequence is a Markov process, so the successive states are independently generated with increasing  $k$  according to

$$\mathbf{x}_k \sim p(\mathbf{x}_k | \mathbf{x}_{k-1}) = p_\theta(\theta_k | \theta_{k-1}, \tau_k, \tau_{k-1}) p_\tau(\tau_k | \theta_{k-1}, \tau_{k-1}). \quad (5.4)$$

These assumptions and models, apart from the constraint  $\tau_k > \tau_{k-1}$ , are very general, and the specific choices are dictated by the application under investigation.

The measurements  $\mathbf{z}_t$ ,  $t \in \mathbb{N}$ , occur on a regular time grid and in the case of the standard PF can be uniquely associated with the corresponding state  $\mathbf{x}_t$ . In the VRPF framework, the underlying state process is asynchronous with the measurement process and the rate of arrival of the measurements is typically (but not necessarily) higher than that of the state process. In order to define the appropriate observation model (also called the likelihood) in this case, where there may be no corresponding state variable for the measurement at time  $t$ , the data points  $\mathbf{z}_t$  are assumed to be

independent of all other data points, conditionally upon the neighborhood  $\mathcal{N}_t$  of states  $\mathbf{x}_{\mathcal{N}_t} = \{\mathbf{x}_k; k \in \mathcal{N}_t\}$ , that is

$$\mathbf{z}_t \sim p(\mathbf{z}_t | \mathbf{x}_0, \dots, \mathbf{x}_\infty) = p(\mathbf{z}_t | \mathbf{x}_{\mathcal{N}_t}). \quad (5.5)$$

The neighborhood  $\mathcal{N}_t$  is constructed as a deterministic function of the time index  $t$  and the state sequence  $\mathbf{x}_{0:\infty}$  and thus it is a random variable itself (this feature is not present in the standard state-space models). For practical (computational) reasons, the neighborhood  $\mathcal{N}_t$  will contain only states whose times  $\tau_k$  are “close” to the observation time  $t$ . Furthermore, the interpolated state  $\hat{\theta}_t = h_t(\mathbf{x}_{\mathcal{N}_t})$  is used, where  $h_t(\cdot)$  is a deterministic function of the state in the neighborhood  $\mathcal{N}_t$ . The observation density (5.5) is then expressed as

$$p(\mathbf{z}_t | \mathbf{x}_{\mathcal{N}_t}) = p(\mathbf{z}_t | \hat{\theta}_t). \quad (5.6)$$

In general, the construction of the state process and the neighborhood structure is not unique and for any given model and different choices will lead to different algorithmic trade-offs.

Having all the definitions, we aim to recursively estimate the sequence of variable-rate state points as new measurements become available. Similarly to the standard PF, the VRPF distribution  $p(\mathbf{x}_{0:\mathcal{N}_t^+} | \mathbf{z}_{0:t})$  can be obtained using the two-step predict-update procedure, similar to (5.3) [56, 57], where  $\mathcal{N}_t^+$  denotes the index of the state variable in  $\mathcal{N}_t$  that has the largest time index  $\tau_k$ . Using the factorization (5.4), we model the MT dynamics with the transition priors

$$\begin{aligned} p_\theta(\theta_k | \theta_{k-1}, \tau_k, \tau_{k-1}) &= p(v_k | v_{k-1}, y_k, y_{k-1}, \tau_k, \tau_{k-1}) p(y_k | v_{k-1}, y_{k-1}, \tau_k, \tau_{k-1}) \\ &= p(v_k | v_{k-1}) \delta(y_k - y_{k-1} - v_{k-1}(\tau_k - \tau_{k-1})), \end{aligned} \quad (5.7)$$

$$p_\tau(\tau_k | \theta_{k-1}, \tau_{k-1}) = \mathcal{U}_{[\tau_{k-1} + \tau^0, \tau_{k-1} + \tau^1]}, \quad (5.8)$$

where  $\mathcal{U}_{[a,b]}$  denotes the uniform distribution in the range  $[a, b]$ . Thus, the states  $\mathbf{x}_k$  for the prediction-update procedure are sampled as

$$\begin{aligned} \tau_k - \tau_{k-1} &\sim \mathcal{U}_{[\tau^0, \tau^1]}, \\ y_k &= y_{k-1} + v_{k-1}(\tau_k - \tau_{k-1}), \\ v_k &\sim p(v_k | v_{k-1}). \end{aligned} \quad (5.9)$$

The sampling of the new states  $\mathbf{x}_k$  at time  $t$  is performed only for those particles  $\mathbf{x}_{k-1}^{(i)}$  for which  $\tau_{k-1}^{(i)} \leq t$ , which also reduces the computational load compared to the standard PF implementation.

The crucial point here is to efficiently model the prior  $p(v_k | v_{k-1})$  in order to catch the rapid changes in edge orientation (corresponding to the state transitions described in Section 5.2.1). The underlying assumption about the MT dynamics in this study is that the MT end can either grow with nearly constant velocity  $\nu^+$ , shrink with nearly constant velocity  $\nu^-$ , or show almost no activity ( $\nu^0 \approx 0$ ). This idealization of reality can be justified by specifying additionally the variances for the velocity estimates,  $\sigma_{\nu^+}^2, \sigma_{\nu^-}^2, \sigma_{\nu^0}^2$ , which account for small deviations in the measured velocities from the

average values  $\nu^+$ ,  $\nu^-$ , and  $\nu^0$ . Taking into account three possible types of motion, we define the following prior  $p(v_k|v_{k-1})$  for the velocity component  $v_k$

$$p(v_k|v_{k-1}) = \begin{cases} (1-a)\mathcal{N}(v_{k-1}, \sigma_{\nu^+}^2) + a\mathcal{N}(\nu^-, \sigma_{\nu^-}^2), & \text{for } v_{k-1} > V_{\text{th}}, \\ (1-a)\mathcal{N}(v_{k-1}, \sigma_{\nu^-}^2) + \\ \quad a(\mathcal{N}(\nu^+, \sigma_{\nu^+}^2) + \mathcal{N}(\nu^0, \sigma_{\nu^0}^2))/2, & \text{for } v_{k-1} < -V_{\text{th}}, \\ (1-a)\mathcal{N}(v_{k-1}, \sigma_{\nu^+}^2) + a\mathcal{N}(\nu^+, \sigma_{\nu^+}^2), & \text{for } |v_{k-1}| < V_{\text{th}}, \end{cases} \quad (5.10)$$

where  $0 < a < 1$  is a weighting parameter that balances the mixture components corresponding to different types of motion in the transition pdf (in tracking applications,  $a$  would correspond to the probability of object/target birth). The threshold  $V_{\text{th}}$  defines which prior should be used: it defines the smallest velocity below which all the small changes in the MT length are considered to belong to state  $S_0$ . Since all three types of MT motion are quite different, the performance of the algorithm is not influenced by possible inaccuracies in setting up the threshold  $V_{\text{th}}$ , which can be estimated in advance from the experimental data. Additionally, the thresholding at  $V_{\text{th}}$  does not imply that at every time point we assume that the system evolves according to only one model. Due to the probabilistic nature of the VRPF, at every time step the posterior pdf describes the probability to find the MT in each of the three states.

In order to define the likelihood  $p(\mathbf{z}_t|\mathbf{x}_{\mathcal{N}_t})$ , we model the edge appearance using an observation model that we have previously used successfully for tracking of tubular structures in noisy medical images [133, 134]. The proposed model describes a small perfectly sharp edge and consist of two rectangular regions,  $S_B$  and  $S_F$  (black and white rectangles in Fig. 5.5, respectively). For each intermediate state  $\hat{\theta}_t = h_t(\mathbf{x}_{\mathcal{N}_t})$ , which is required for the likelihood computation, the neighborhood is defined as  $\mathcal{N}_t = \{k, k-1; \tau_{k-1} \leq t < \tau_k\}$ . For the MT length changes, linear interpolation between two neighboring states  $\theta_k$  and  $\theta_{k-1}$  is used,  $y_t = y_{k-1} + v_{k-1}(t - \tau_{k-1})$ , and the orientation of the rectangles for each time point  $t$  is defined by the velocity component  $v_{k-1}^{(i)}$ . The regions  $S_B$  and  $S_F$  are defined as follows

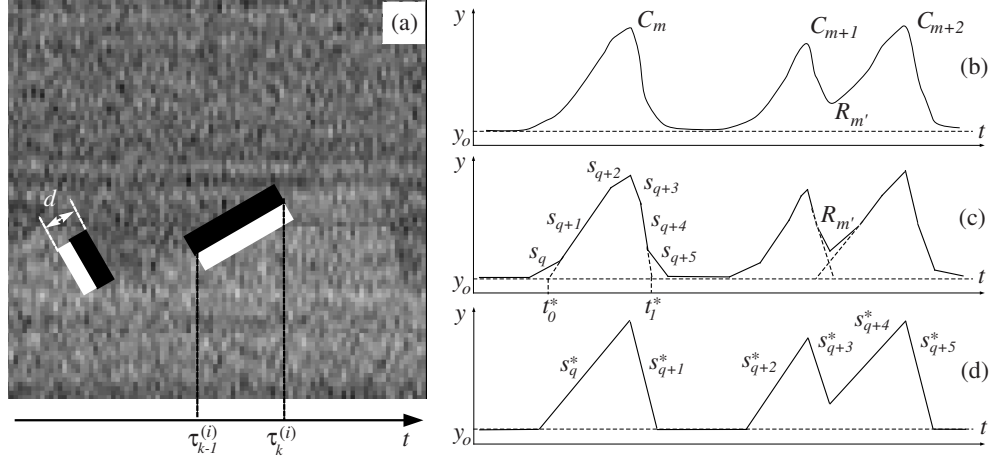
$$S_B(\hat{\theta}_t) = S_B(\tau_{k-1}, \tau_k, v_{k-1}) = \left\{ \left( \frac{l - v_{k-1}b}{\sqrt{1 + v_{k-1}^2}}, \frac{lv_{k-1} + b}{\sqrt{1 + v_{k-1}^2}} \right) : l \in [0, l_v], b \in [0, d] \right\},$$

$$S_F(\hat{\theta}_t) = S_F(\tau_{k-1}, \tau_k, v_{k-1}) = \left\{ \left( \frac{l + v_{k-1}b}{\sqrt{1 + v_{k-1}^2}}, \frac{lv_{k-1} - b}{\sqrt{1 + v_{k-1}^2}} \right) : l \in [0, l_v], b \in [0, d] \right\},$$

where  $l_v = (\tau_k - \tau_{k-1})\sqrt{1 + v_{k-1}^2}$ .

To measure the likelihood of edge existence at some image position with an orientation defined by the velocity component of the state vector, the average image intensity values,  $\mu_B$  and  $\mu_F$ , are computed over the regions  $S_B$  and  $S_F$ . The likelihood is defined as

$$p(\mathbf{z}_t|\mathbf{x}_{\mathcal{N}_t}) = \begin{cases} \exp\left(\frac{\mu_F - \mu_B}{\gamma}\right) - 1, & \mu_F - \mu_B > 0, \\ 0, & \mu_F - \mu_B \leq 0, \end{cases} \quad (5.11)$$



**Figure 5.5.** The observation model used in the experiments, which compares the intensity distribution in two rectangular regions (black and white strips) and defines the likelihood of edge existence (a). Examples of applying the MTA to the extracted edge using the VRPF in order to compute the kinematic parameters (b-d).

which defines the pdf of the edge location and favors sharp edges over smoother noisy intensity transitions. Two model parameters that control the sensitivity to the edge location, the width  $d$  and the scaling factor  $\gamma$ , should be specified. The length  $l_v$  is automatically defined by the time sampling functions (5.8). The variety in the length of the observation model adds a multiscale property to the analysis. In general, for small values of  $l_v$  the estimation of  $\mu_B$  and  $\mu_F$  is less accurate than for larger values of  $l_v$ . Additionally, for large  $l_v$  the sensitivity of the observation model to the edge orientation increases – the likelihood decreases rapidly for small misalignments of the observation model with the edge. Usually this is a desirable property, because the edge can be located more precisely. The disadvantage of using only large  $l_v$  is the disability of the observation model to capture the fast motion transition stages.

Alternatively, the gradient image can be used as measurements for the VRPF, which represents the edges computed using the Gaussian derivatives. In this case, the pixel value at some position in the gradient image is the likelihood for finding the edge. Depending on the scale at which the derivatives were computed, the slopes of the tangent lines, which are related to the velocity values, can be accurately estimated, but only in regions having the same motion type. It can be seen from Fig. 5.4 that in the regions of the gradient image where catastrophes are present, the edge appearance is distorted – the transition between the growth and shrinkage is smoothed. This leads to a lowering of the angles of the tangent lines and, as a result, to underestimation of the velocity values. Due to the mentioned nonlinearity, this underestimation is especially severe for the shrinkage velocity.

In order to derive the MMSE estimator, the principle of fixed-lag smoothing is used, which greatly improves the final results. Here, the MMSE estimate of the state



at time  $t - \Delta t$  is computed using the posterior as distribution  $p(\mathbf{x}_{0:N_t^+} | \mathbf{z}_{0:t})$ , that is

$$\hat{y}_{t-\Delta t} = \sum_{i=1}^{N_s} w_t^{(i)} h_t(\mathbf{x}_{N_t-\Delta t}). \quad (5.12)$$

In other words, the estimation of the edge position at time  $t$  is delayed until the measurements at time  $t + \Delta t$  will be processed and the posterior updated.

### 5.2.5 Multiscale Trend Analysis

Having the estimated edge  $\hat{y}_t$  after applying the VRPF, we employ multiscale trend analysis (MTA) [188] in order to automatically compute all the parameters of interest. At this stage of our analysis, it is necessary to detect all the catastrophe and rescue events and split the live history  $\hat{y}_t$  into parts of growth and shrinkage, possibly separated by stages of no activity (state  $S_0$ ),

The MTA was originally proposed for analysis of trends in time series and was recently successfully applied for analysis of MT transport in melanophores [189]. Compared to methods that try to construct an optimal piecewise linear approximation  $L_\epsilon(t)$  with a minimal number of segments for a given error  $\epsilon$ , the MTA builds a multi-level hierarchy of consecutively more detailed piecewise linear approximations of the analyzed time series at different scales. In general, it is not known beforehand which scale should be used for the analysis, but some prior knowledge about the application can significantly narrow down the range of levels that should be analyzed after applying MTA.

The following robust procedure was experimentally found to produce accurate estimates of the kinematic parameters using MTA. First, MTA decomposition is performed for a number of levels,  $l = \{1, \dots, N_L\}$ , where  $N_L$  is a fixed (large) number. Each level in the decomposition can be represented with a set of nodes  $\{s_q\}_{q=1}^l$  that partition  $\hat{y}(t)$  on the interval  $[0, T]$ , where each node is given by four parameters,  $(t_0^q, t_1^q, \alpha^q, \tilde{y}^q)$ , and describes the linear approximation of  $\hat{y}(t)$  on the interval  $[t_0^q, t_1^q]$  with slope  $\alpha^q$  and intercept  $\tilde{y}^q = \hat{y}(t_0^q)$ . In our implementation of MTA, the number of nodes (piecewise linear approximations) at level  $l$  is equal to  $l$ , and the first level ( $l = 1$ ) is given by the base line  $y = y_0$ , where  $y_0 = \min_t \hat{y}(t)$ . At each level  $l$ , the number of catastrophes (local maxima in the approximation of  $\hat{y}(t)$  at that level)  $N_{\text{cat}}(l)$ , is computed. Due to the nature of the signal  $\hat{y}(t)$  and the way MTA works, for some range of hierarchy levels the number of catastrophes will stay constant ( $dN_{\text{cat}}/dl = 0$ ). In general, the function  $N_{\text{cat}}(l)$  is non-decreasing. By finding the maximum in the histogram of  $\{N_{\text{cat}}(l) : l = \{1, \dots, N_L\}\}$ , which shows how many levels contain the same number of catastrophes, we can obtain the number of actual catastrophe events  $N_{\text{cat}}^*$ . From the set of levels  $\{l_j\}$  that correspond to  $N_{\text{cat}}^*$  (satisfying  $N_{\text{cat}}(l_j) = N_{\text{cat}}^*$ ), the median is selected,  $l^*$ , as the level for further parameter computations.

For the selected decomposition level and each catastrophe event  $C_m$ ,  $m = \{1, \dots, N_{\text{cat}}^*\}$ , which occurs at time  $t_m^c$ , the two sets of neighboring nodes,  $\{s_q : t_{m-1}^c < t_0^q < t_m^c \cap \alpha^q > 0, q = 1, \dots, l^*\}$  and  $\{s_q : t_m^c < t_1^q < t_{m+1}^c \cap \alpha^q < 0, q = 1, \dots, l^*\}$  are analyzed (see Fig. 5.5(c)), where  $t_0^c = 0$  and  $t_{N_{\text{cat}}^*+1}^c = T$ . On both sides of the local maximum  $C_m$ , the nodes with the steepest slope  $\alpha^q$  are selected and the linear



approximations corresponding to those nodes are extrapolated until the intersection with  $y = y_0$ , giving the values  $\tilde{t}_m^0$  and  $\tilde{t}_m^1$ . The rescue event  $R_{m'}$  ( $m' \in \mathbb{N}$ ) is detected between two catastrophes  $C_m$  and  $C_{m+1}$  if  $\tilde{t}_m^1 > \tilde{t}_{m+1}^0$ . In this case, the local minimum in the approximation of  $\hat{y}(t)$  on the interval  $[\tilde{t}_{m+1}^0, \tilde{t}_m^1]$  gives the position of the rescue,  $t_{m'}^R$ . Then, the approximation is recomputed for  $\hat{y}(t)$  on the intervals  $[\tilde{t}_m^0, t_m^c]$  and  $[t_m^c, \tilde{t}_m^1]$ . If the rescue event is positioned between two catastrophes  $C_m$  and  $C_{m+1}$ , the approximation is recomputed on the interval  $[t_m^c, t_{m'}^R]$ . The new approximation is given by a new set of nodes  $S^* = \{s_q^*\}_{q=1}^{2N_{\text{cat}}^*}$  (see Fig. 5.5(d)), which is used to compute the kinematic parameters: the total growth and shrinkage times ( $T^+$ ,  $T^-$ ) and the corresponding velocity ( $\hat{\nu}^+$ ,  $\hat{\nu}^-$ ) and frequency  $f_{\text{cat}}$  and  $f_{\text{res}}$  estimates:

$$T^+ = \sum_{\substack{\forall s_q^* \in S^* \\ \alpha^q > 0}} (t_1^q - t_0^q), \quad \hat{\nu}^+ = \frac{1}{T^+} \sum_{\substack{\forall s_q^* \in S^* \\ \alpha^q > 0}} (t_1^q - t_0^q) \alpha^q, \quad (5.13)$$

$$T^- = \sum_{\substack{\forall s_q^* \in S^* \\ \alpha^q < 0}} (t_1^q - t_0^q), \quad \hat{\nu}^- = \frac{1}{T^-} \sum_{\substack{\forall s_q^* \in S^* \\ \alpha^q < 0}} (t_1^q - t_0^q) \alpha^q, \quad (5.14)$$

$$f_{\text{cat}} = N_{\text{cat}}^*/T^+, \quad f_{\text{res}} = N_{\text{res}}^*/T^-, \quad (5.15)$$

where  $N_{\text{res}}^*$  is the number of rescue events. In practice, the VRPF outputs a good piecewise linear approximation of the edges, so that the described procedure based on MTA runs robustly and accurately.

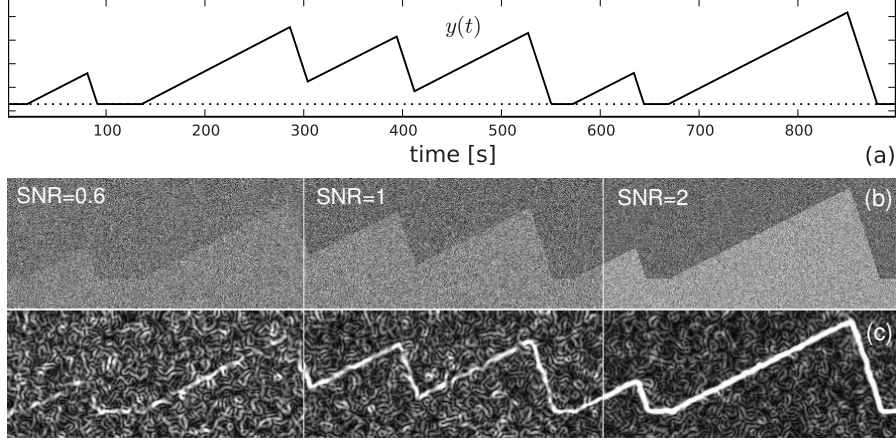
## 5.3 Experimental Results

The performance of the proposed VRPF-based method was evaluated using both synthetic images (Section 5.3.1) and real data from studies of MT dynamics *in vitro* (Section 5.3.2). The synthetic images, for which the ground truth was available, gave the possibility to explore the accuracy and robustness of the method depending on the image quality (different SNR levels) and the parameter values that model the MT dynamics. The experiments on real data enabled us to compare the estimated kinematic parameters with manual analysis by expert biologists.

### 5.3.1 Evaluation on Synthetic Data

#### 5.3.1.1 Simulation Step

The proposed technique was evaluated using computer generated kymoimages for different SNRs. The dynamics of the MT tip was simulated according to the model described in Section 5.2.1 (Fig. 5.1). The values of the model parameters were randomly generated each time the MT changes its state, by drawing a sample from the Gamma distribution,  $\tau \sim \mathcal{G}(4, 1)$ , and, depending on which state the MT is entering, the duration times were defined as  $\tau^+ = 20\tau$ ,  $\tau^- = 10\tau$ ,  $\tau^0 = 10\tau$ . The corresponding velocity values were drawn from the Gaussian distribution,  $\nu^+ \sim \mathcal{N}(0.5, 0.005)$ ,



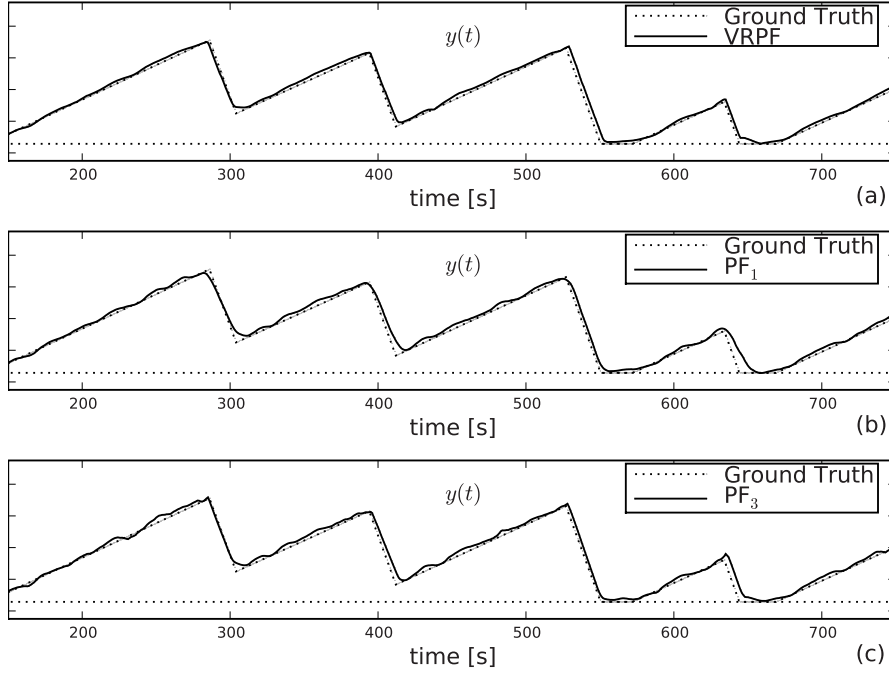
**Figure 5.6.** Examples of the synthetic images used in the experiments. The simulated MT tip dynamics (a) is used to create the synthetic images for different SNR levels (b), for which the gradient images (c) are computed using the Gaussian derivatives at scale  $\sigma_G = 3$ .

$\nu^- \sim \mathcal{N}(-3, 0.005)$ ,  $\nu^0 \sim \mathcal{N}(0, 0.05)$ . These model values are representative of practical values.

Having the simulated dynamics  $y(t)$ ,  $0 < t < T$  (see Fig. 5.6(a) for an example), we created corresponding images of size  $T \times Y$ , where  $T = 1000$  and  $Y = \max_t y(t) + 2y_0$  for several SNR levels. Padding with a strip of size  $T \times y_0$ ,  $y_0 = 20$ , was applied to the top and bottom of the image to avoid border problems when using the described rectangular observation model (Section 5.2.4). The height of the generated images was in the range of 100 – 150 pixels, which corresponds to 8 – 12  $\mu\text{m}$  ( $\Delta_t = 1\text{s}$  and  $\Delta_y = 80\text{ nm}$ ). For all  $t$ , the image pixels were filled with background intensity  $I_B = 100$  if  $j > y(t) + y_0$  and otherwise with foreground intensity  $I_F = I_B + \sigma\text{SNR}$ , where  $\sigma = 10$ . To create the final noisy image, each pixel value was replaced with a random sample from the distribution  $\mathcal{N}(I(t, j), \sigma^2)$ . For the chosen values of  $I_B$  and  $\sigma = 10$ , this corresponds to the Poisson noise model, which is dominant in images obtained using light microscopy [185]. Examples of synthetic images for various SNRs are shown in Fig. 5.6(b). Again, for visual comparison, the edge information (the gradient magnitude) obtained using the Gaussian derivatives at scale  $\sigma_G = 3$  is shown in Fig. 5.6(c).

The parameters of the described VRPF algorithm were fixed to the following values:  $\nu^+ = 0.5$ ,  $\nu^- = -3$ ,  $\sigma_{\nu^0}^2 = 0.5$ ,  $\sigma_{\nu^+}^2 = 0.05$ ,  $\sigma_{\nu^-}^2 = 0.5$ ,  $V_{\text{th}} = 0.15$ ,  $d = 6$ ,  $\tau^0 = 3$ ,  $\tau^1 = 10$ ,  $\Delta t = 20$ ,  $N_s = 500$ ,  $N_L = 80$ ,  $a = 0.01$ ,  $\gamma = 10$ . Since the ground truth was available in these experiments, the accuracy of extracting the edges was evaluated using a traditional quantitative performance measure: the root mean square error (RMSE) [104]:

$$\text{RMSE} = \sqrt{\frac{1}{|T|} \sum_{t \in T} (y_t - \hat{y}_t)^2}, \quad (5.16)$$

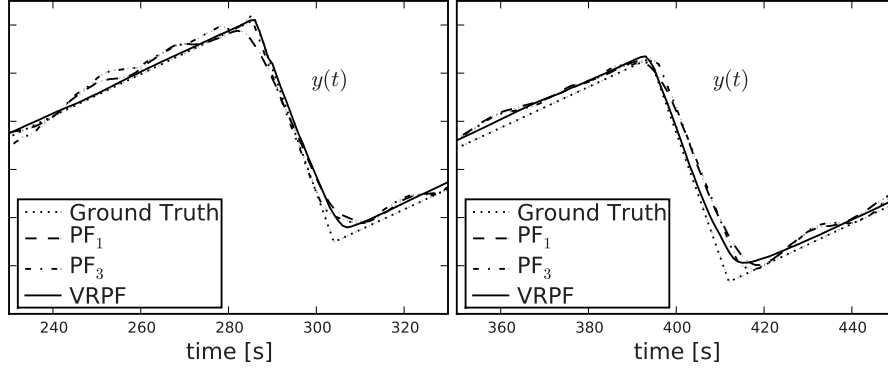


**Figure 5.7.** Sample results of extracting the edge information from the noisy synthetic images using the proposed VRPF and two types of standard PFs in comparison with the ground truth.

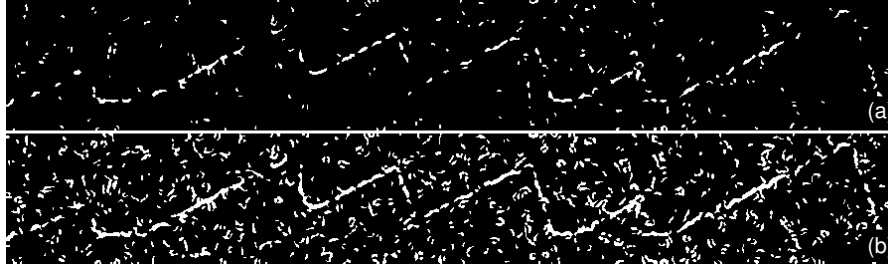
where  $y_t$  defines the true position of the edge at time  $t$ ,  $\hat{y}_t$  is a MMSE estimate of  $y_t$  given by the VRPF,  $\mathcal{T}$  is the set of time points for which the edge exists, and  $|\cdot|$  denotes the set size operator.

### 5.3.1.2 Results

The proposed VRPF method was evaluated using 20 synthetically generated images. Examples of edge extraction for  $\text{SNR} = 0.6$  are shown in Fig. 5.7. In addition to the proposed VRPF, we also implemented two standard particle filters, denoted  $\text{PF}_1$  and  $\text{PF}_3$ , in which the state transition process is synchronous with the measurement process (see Section 5.2.4).  $\text{PF}_1$  uses only one state transition model,  $p(\mathbf{x}_t|\mathbf{x}_{t-1})$ , which describes nearly-constant velocity motion [141]. To capture abrupt changes in the edges, the variance of the process noise in this transition model had to be made rather large. Due to this high variance, the typical overshoots just after the catastrophe events (see Fig. 5.7(b)) highly corrupted the slope estimates, in particular the estimation of the shrinkage velocity. Additionally, for the low SNR image data, the filter frequently lost the edge and traced spurious background structures.  $\text{PF}_3$  uses the same set of transition models as the VRPF. Contrary to the observation model used in the VRPF, however, a rectangular observation model of the same width  $d$



**Figure 5.8.** More detailed results of extracting the edge information from the noisy synthetic images using the proposed VRPF and two types of standard PFs in comparison with the ground truth. The plots are zooms of the first two peaks in Fig. 5.7 and show the results combined.



**Figure 5.9.** Results of edge extraction using the Canny edge detector for two different values of the hysteresis thresholds.

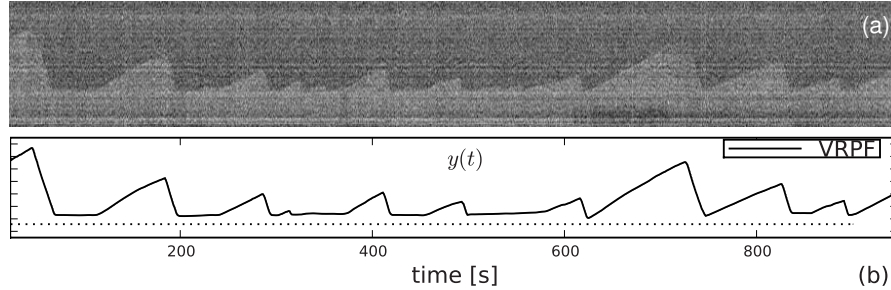
but fixed length  $l_v = 5$  was used. The zoomed results in Fig. 5.8 clearly show that the edge  $\hat{y}(t)$  estimated using the standard PFs is typically less smooth and piecewise linear. For visual comparison, the edge information extracted using the Canny edge detector [25] for two different values of hysteresis thresholds is shown in Fig. 5.9.

The results of applying MTA for kinematic parameter estimation based on the edges extracted using  $\text{PF}_3$  and VRPF are shown in Table 5.1 (results for  $\text{PF}_1$  are not given here, since this filter frequently failed to find the edges at all, as indicated above). The RMSEs for both  $\text{PF}_3$  and VRPF in finding the edge are approximately the same, but the velocity estimates computed using the linear approximation are different. This difference depends on the absolute value of the velocity, and for higher velocity values (especially the shrinkage velocity), VRPF is about 3-7% more accurate than  $\text{PF}_3$ . The results also show that prefiltering of the images does not improve the estimates significantly. This indicates that the observation model robustly estimates the mean intensities in the regions  $S_B$  and  $S_F$  even at very low SNRs. Prefiltering in this case worsens the estimation by blurring the already hardly visible edges before applying the VRPF.

**Table 5.1.** Results of parameter estimation in synthetically generated images of microtubule dynamics using MTA based on the edges extracted with different combinations of prefiltering and particle filtering methods.

SNR	RMSE	$\nu^+ \pm \text{sd}$	$\nu^- \pm \text{sd}$	$f_{\text{cat}}$	$f_{\text{res}}$
Ground truth values					
-	-	$0.50 \pm 0.005$	$-3.00 \pm 0.005$	0.009	0.018
VRPF without prefiltering					
0.4	2.54	$0.47 \pm 0.07$	$-2.41 \pm 0.79$	0.011	0.019
0.6	1.43	$0.50 \pm 0.03$	$-3.03 \pm 0.61$	0.009	0.018
0.8	1.23	$0.49 \pm 0.02$	$-2.91 \pm 0.62$	0.009	0.017
1.0	1.15	$0.50 \pm 0.01$	$-2.96 \pm 0.37$	0.009	0.017
1.2	0.96	$0.49 \pm 0.01$	$-2.95 \pm 0.34$	0.009	0.018
VRPF with bilateral prefiltering					
0.4	2.01	$0.48 \pm 0.07$	$-2.44 \pm 0.83$	0.010	0.017
0.6	1.86	$0.50 \pm 0.02$	$-2.86 \pm 0.40$	0.009	0.015
0.8	1.64	$0.49 \pm 0.02$	$-2.93 \pm 0.34$	0.009	0.017
1.0	1.33	$0.49 \pm 0.03$	$-3.05 \pm 0.36$	0.009	0.017
1.2	1.25	$0.49 \pm 0.02$	$-2.98 \pm 0.32$	0.009	0.018
VRPF with anisotropic diffusion prefiltering					
0.4	2.41	$0.47 \pm 0.08$	$-2.14 \pm 0.56$	0.010	0.019
0.6	2.55	$0.49 \pm 0.08$	$-2.91 \pm 0.64$	0.010	0.021
0.8	1.44	$0.49 \pm 0.03$	$-2.98 \pm 0.39$	0.009	0.018
1.0	1.13	$0.49 \pm 0.02$	$-2.91 \pm 0.44$	0.009	0.018
1.2	1.05	$0.49 \pm 0.02$	$-2.91 \pm 0.34$	0.009	0.018
PF <sub>3</sub> without prefiltering					
0.4	2.72	$0.47 \pm 0.08$	$-2.44 \pm 1.02$	0.006	0.026
0.6	1.46	$0.50 \pm 0.05$	$-2.71 \pm 0.92$	0.011	0.014
0.8	1.12	$0.50 \pm 0.05$	$-2.73 \pm 0.21$	0.009	0.017
1.0	0.98	$0.49 \pm 0.02$	$-2.81 \pm 0.27$	0.009	0.015
1.2	1.02	$0.49 \pm 0.02$	$-2.79 \pm 0.31$	0.009	0.018

We also assessed the sensitivity of the proposed VRPF method to changes in the expected velocities. To this end, the parameter values  $\nu^+$  and  $\nu^-$  were varied. It was observed that deviation of these parameters from the ground truth values decreased the accuracy of the method. In practice, however, this inaccuracy can be easily reduced, by running the algorithm iteratively, in a “bootstrapping” fashion. First, the initial velocity values  $\nu^+$  and  $\nu^-$  are approximately specified, with large standard deviations  $\sigma_{\nu^+}$  and  $\sigma_{\nu^-}$ . After the first run, these parameters, which are still inaccurate but now closer to the optimal values, are reestimated. Then, the algorithm is initialized with the new estimates and rerun. In the experiments, we found that this approach always resulted in estimates in the range  $(\nu \pm \sigma_\nu)$  defined by the ground truth.



**Figure 5.10.** Example of a kymograph generated in the experiments on real DIC microscopy image data with  $\text{SNR} \approx 1$  (a) and the results of applying the proposed VRPF (b).

### 5.3.2 Evaluation on Real Data

For the validation on real data we collected three representative DIC microscopy image sequences acquired to study the influence of different concentrations of EB3 (end-binding protein 3) and GFP-EB3 (EB3 fused to the green fluorescent protein) on the MT growth and shrinkage velocities ( $\nu^+$  and  $\nu^-$ ) and the catastrophe rate ( $f_{\text{cat}}$ ). The sequences were taken from experiments with MT nucleation from stable tubulin seeds, where  $15\mu\text{M}$  of tubulin was added (Experiment I), or, in addition,  $1\mu\text{M}$  of EB3 (Experiment II), or  $1\mu\text{M}$  of GFP-EB3 (Experiment III) [78]. From each sequence, 10-20 MTs were selected by biologists, and the observation lines were drawn manually. The image sequences contained about 1000-1200 frames (one per second) of size  $700 \times 500$  pixels (of size  $86 \times 86 \text{ nm}^2$ ). To estimate the parameters of interest, for each experiment 10 kymographs were constructed and analyzed manually and using the proposed VRPF method. The results are presented in Table 5.2, where the speed estimates are also converted to  $\mu\text{m}/\text{min}$ . The usage of these units is more common in biological experiments and it also allows straightforward comparison with the recently published results [78]. An example of edge extraction using VRPF in real data is shown in Fig. 5.10. A comparison of the estimates obtained by manual and VRPF-based analysis suggests that the proposed automatic method may replace the laborious manual procedures.

## 5.4 Discussion and Conclusions

In this chapter we have proposed a new approach for the automatic analysis of *in vitro* microtubule dynamics imaged using time-lapse differential interference contrast microscopy. It is based on a transformation of the 2D image sequences into kymographs (space-time images) for each microtubule along a corresponding observation line. By using this representation, the task of tracking microtubule tips on a per-frame basis in the noisy images, which from our previous work is known to be a difficult and error-prone problem, is replaced by a segmentation of spatiotemporal structures (edges in our case). For the extraction of these structures from the kymographs, we have proposed a variable-rate particle filtering method, which is better capable of

**Table 5.2.** Results of parameter estimation in real DIC microscopy image data sets using manual analysis versus VRPF.

	$\nu^+ \pm \text{sd}$ [pix/frame]	$\nu^- \pm \text{sd}$ [pix/frame]	$f_{\text{cat}}$	$\nu^+$ [ $\mu\text{m}/\text{min}$ ]	$\nu^-$ [ $\mu\text{m}/\text{min}$ ]
	Experiment I (pure tubulin)				
Manual	0.19 $\pm$ 0.04	-2.06 $\pm$ 0.43	0.0021	0.56	-10.63
VRPF	0.17 $\pm$ 0.07	-1.89 $\pm$ 0.52	0.0020	0.51	-9.72
	Experiment II (tubulin and EB3)				
Manual	0.52 $\pm$ 0.05	-2.78 $\pm$ 0.65	0.0133	2.68	-14.34
VRPF	0.49 $\pm$ 0.07	-2.84 $\pm$ 0.51	0.0141	2.52	-14.65
	Experiment II (tubulin and GFP-EB3)				
Manual	0.49 $\pm$ 0.08	-2.88 $\pm$ 0.41	0.0132	2.52	-14.86
VRPF	0.50 $\pm$ 0.06	-2.72 $\pm$ 0.50	0.0145	2.58	-14.03

dealing with abrupt changes than standard particle filtering methods. The method is built within a Bayesian framework and optimally combines the measurements and prior knowledge about the underlying processes. For the estimation of important kinematic parameters from the extracted edges, we have adopted multiscale trend analysis.

The quantitative evaluation of the proposed method was done using realistic synthetic images as well as real microscopy image data from biological experiments. From the results of the experiments on synthetic data, where the ground truth of the microtubule tip position was available, it was concluded that the method is capable of accurate estimation of the important kinematic parameters. Moreover, it was concluded that the method is more robust and more accurate than standard particle filtering methods. For the real data, the proposed method was compared to manual analysis carried out by expert biologists. The results of this comparison clearly demonstrated that the automatically estimated parameters are in good agreement with the results obtained manually. Together, these observations lead to the conclusion that the proposed method may replace laborious manual analyses.



---

## Summary

---

*If people do not believe that mathematics is simple, it is only because they do not realize how complicated life is.*

— JOHN LOUIS VON NEUMANN (1903–1957)

Advances in fluorescent probing and microscopic imaging technology have revolutionized biology in the past decade and have opened the door for studying subcellular dynamical processes. A simple description of sometimes complex patterns of movement in living cells may give insight in the underlying mechanisms governing these movements. However, accurate and reproducible methods for processing and analyzing the images acquired for such studies are still lacking. Since manual image analysis is time consuming, potentially inaccurate, and poorly reproducible, many biologically highly relevant questions are either left unaddressed, or are answered with great uncertainty. Hence, the development of automated image analysis techniques for accurate and reproducible tracking and motion analysis of subcellular structures from time-lapse microscopy image data is crucial.

Recent results in psychophysics and human vision research have revealed the highly integrated nature of vision systems in using spatial, temporal, and prior information [23]. Local motion signals are often ambiguous, and many important motion phenomena can be explained by hypothesizing that the human visual system uses temporal coherence to resolve ambiguous inputs. It has therefore been proposed that input data are temporally grouped and used to predict and estimate the motion flows in image sequences. Such temporal grouping can be expressed in terms of a Bayesian generalization of standard Kalman filtering. Existing tracking techniques, whether commercial or academic, generally make very limited use of temporal information and prior knowledge.

The subject of this thesis is particle filtering methods and their application for multiple object tracking in different biological imaging applications. Particle filtering (PF) is a technique for implementing recursive Bayesian filtering by Monte Carlo sampling. A fundamental concept behind the Bayesian approach for performing inference is the possibility to encode the information about the imaging system, possible noise sources, and the system dynamics in terms of probability densities. Nevertheless, the Bayesian tracking framework is rather a “recipe” than a ready-to-use solution to a given problem, which should be implemented in practice, for example using the

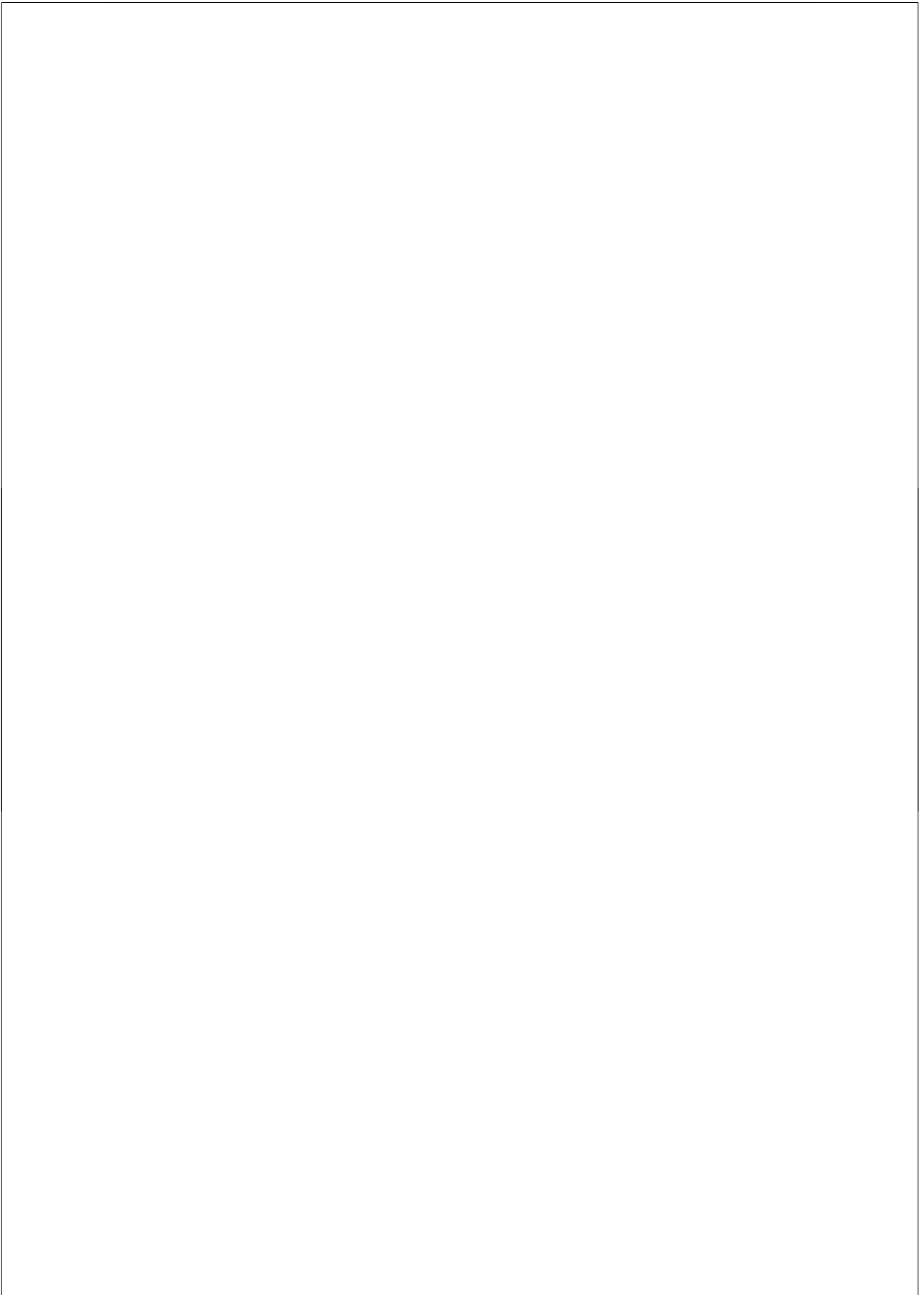
PF approximation. In general, the construction of particle filters is not unique, and for any given application will lead to different algorithmic trade-offs. As stated by some methodologists in this field “the design of efficient particle filters is still more of an art than a science” [126]. In this thesis, a set of novel PF based methods for sub-cellular motion analysis is developed. The applicability of these methods for robust and accurate detection and tracking of large numbers of small objects in 2D and 3D image sequences obtained by fluorescence microscopy imaging as well as for dynamics analysis using kymographs is demonstrated and evaluated.

Contrary to the conventional two-step (detection and linking) approaches to tracking, Bayesian tracking does not require a separate object detection procedure. Nevertheless, robust and accurate detectors can be efficiently used in the Bayesian framework, especially to initiate new tracks and terminate existing ones, when the object appears in or disappears from the field of view, respectively. In Chapter 2, the performance of six unsupervised and two supervised (machine learning) detection methods for the detection of small spots in fluorescence microscopy images is quantitatively evaluated. It is shown that overall, the supervised methods (AdaBoost and Fisher discriminant analysis) perform better, in that they show the highest true positive rate (at very low false positive rate) and the lowest sensitivity to parameter changes, for all types of image data considered. Nevertheless, the differences in performance are not large compared to some of the unsupervised methods, especially the so-called *h*-dome detector (HD) proposed in the chapter. Based on extensive experiments, the conclusion is drawn that when a detector with overall good performance is needed, the mentioned supervised detectors or the unsupervised HD detector are to be preferred. The disadvantage of the supervised methods is that they rely on a training stage, which involves the extraction of positive and negative samples from the image data beforehand. This requires manual annotation of thousands of objects in order to achieve sufficient discriminating power, and is extremely tedious, time consuming, and observer dependent. Taking this into account, the unsupervised HD detector is much easier to use in practice. Finally, when the SNR is sufficiently high ( $> 5$  as a rule of thumb), the other unsupervised detectors perform just as well, and require only minimal adjustment of their parameters to the specific application.

Chapter 3 describes the derivation of a novel particle filter for quantitative analysis of subcellular dynamics, in this case for microtubule growth analysis. The algorithm exploits prior information about the microtubule dynamics and imaging process, which makes it perform superior in the presence of severe noise in comparison with existing frame-by-frame approaches, which break down at  $\text{SNR} < 4-5$  [26,32]. Additionally, the algorithm naturally deals with photobleaching effects. Experiments on synthetic data confirm that the proposed PF yields reliable tracking results even in data with SNR as low as 2, contrary to two other popular tracking tools, and that it is potentially more accurate than manual tracking by expert human observers. Applied to real fluorescence microscopy image sequences from microtubule dynamics, the algorithm performs comparable to human observers. This is explained by the fact that the latter experiments were limited to comparing distributions and averages, which may conceal small local discrepancies, especially when the objects’ velocities vary over time. Instant velocities were also analyzed per track but could not be quantitatively validated due to the lack of ground truth.

Since common Bayesian tracking algorithms are designed to deal with only one specific type of motion, they may fail when used for biological applications where more complex motion patterns need to be analyzed. Therefore, in Chapter 4, the algorithm developed in Chapter 3 is extended to be capable of tracking different types of subcellular objects with different types of motion patterns. The tracking accuracy is improved 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. The proposed algorithm is 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 synthetic data experiments clearly show the superiority of the proposed algorithm over manual tracking as well as previous Bayesian tracking approaches, which were already demonstrated to be superior to alternative non-Bayesian tracking algorithms. The real-data experiments confirm the validity of the tracking results produced by the proposed algorithm. Based on these results, the algorithms are now being explored in practice for addressing specific biological questions.

Finally, Chapter 5 demonstrates the applicability of PF methods to another biological application: the analysis of microtubule dynamics *in vitro*, imaged using differential interference contrast microscopy. A novel algorithm is proposed that combines variable-rate particle filtering (VRPF) and multiscale trend analysis for analyzing the motion of the growing or shrinking tips of microtubules in 2D spatiotemporal images (kymographs). The proposed VRPF optimally combines image information and prior knowledge about the underlying microtubule dynamics and is capable of following the microtubule end even in situations where rapid motion changes (after rescue or catastrophe) occur. As demonstrated by experiments on synthetic data, the method is capable of accurate estimation of the important kinematic parameters. In these experiments, the error of locating the microtubule tip in kymographs is measured and its influence on the estimation of the important kinematic parameters (growth and shrinkage velocities, rescue and catastrophe frequencies) is studied. From theoretical considerations it is known that even relatively small errors in tip localization can lead to large errors in the final parameter estimates, due to the nonlinear relationship between the estimated slopes and the computed velocities. Indeed, the experimental results show increased uncertainty in velocity estimation for higher velocities. Applied to real data, the proposed method produces parameter estimates in accordance with estimates obtained manually by expert biologists.



---

# Bibliography

---

*One's work may be finished someday, but one's education, never.*

— ALEXANDRE DUMAS, PÈRE (1802 - 1870)

- [1] M. D. Abràmoff, P. J. Magalhães, S. J. Ram, “Image processing with ImageJ”, *Biophotonics International*, vol. 11, no. 7, pp. 36–42, 2004.
- [2] F. Aguet, D. Van De Ville, M. Unser, “A maximum-likelihood formalism for sub-resolution axial localization of fluorescent nanoparticles”, *Optics Express*, vol. 13, pp. 10503–10522, 2005.
- [3] W. M. Ahmed, S. J. Leavesley, B. Rajwa, M. N. Ayyaz, A. Ghafoor, J. P. Robinson, “State of the art in information extraction and quantitative analysis for multimodality biomolecular imaging”, *Proceedings of the IEEE*, vol. 96, no. 3, pp. 512–531, 2008.
- [4] A. Akhmanova & C. C. Hoogenraad, “Microtubule plus-end-tracking proteins: Mechanisms and functions”, *Current Opinion in Cell Biology*, vol. 17, no. 1, pp. 47–54, 2005.
- [5] A. Akhmanova, C. C. Hoogenraad, K. Drabek, T. Stepanova, B. Dortland, T. Verkerk, W. Vermeulen, B. M. Burgering, C. I. de Zeeuw, F. Grosveld, N. Galjart, “Clasps are CLIP-115 and -170 associating proteins involved in the regional regulation of microtubule dynamics in motile fibroblasts”, *Cell*, vol. 104, no. 6, pp. 923–935, 2001.
- [6] A. Akhmanova & M. O. Steinmetz, “Tracking the ends: A dynamic protein network controls the fate of microtubule tips”, *Nature Reviews Molecular Cell Biology*, vol. 9, no. 4, pp. 309–322, 2008.
- [7] F. J. Anscombe, “The transformation of Poisson, binomial and negative-binomial data”, *Biometrika*, vol. 35, no. 3/4, pp. 246–254, 1948.
- [8] J. Apgar, Y. Tseng, E. Fedorov, M. B. Herwig, S. C. Almo, D. Wirtz, “Multiple-particle tracking measurements of heterogeneities in solutions of actin filaments and actin bundles”, *Biophysical Journal*, vol. 79, no. 2, pp. 1095–1106, 2000.
- [9] S. M. Arulampalam, S. Maskell, N. Gordon, T. Clapp, “A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking”, *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 174–188, 2002.
- [10] C. P. Bacher, M. Reichenzeller, C. Athale, H. Herrmann, R. Eils, “4-D single particle tracking of synthetic and proteinaceous microspheres reveals preferential movement of nuclear particles along chromatin-poor tracks”, *BMC Cell Biology*, vol. 5, no. 45, pp. 1–14, 2004.
- [11] Y. Bar-Shalom, X. R. Li, T. Kirubarajan, *Estimation with Applications to Tracking and Navigation*, Wiley, New York, 2001.
- [12] J. E. Bear, T. M. Svitkina, M. Krause, D. A. Schafer, J. J. Loureiro, G. A. Strasser, I. V. Maly, O. Y. Chaga, J. A. Cooper, G. G. Borisov, F. B. Gertler, “Antagonism between Ena/VASP proteins and actin filament capping regulates fibroblast motility”, *Cell*, vol. 109, no. 4, pp. 509–521, 2002.

- [13] C. B. Bergsma, G. J. Streekstra, A. W. M. Smeulders, E. M. M. Manders, "Velocity estimation of spots in three-dimensional confocal image sequences of living cells", *Cytometry*, vol. 43, no. 4, pp. 261–272, 2001.
- [14] Peter Bieling, Liedewij Laan, Henry Schek, Laura E. Munteanu, Linda Sandblad, Marileen Dogterom, Damian Brunner, Thomas Surrey, "Reconstitution of a microtubule plus-end tracking system *in vitro*", *Nature*, vol. 450, pp. 1100–1105, December 2007.
- [15] S. Blackman & R. Popoli, *Design and Analysis of Modern Tracking Systems*, Artech House, Norwood, MA, 1999.
- [16] Y. Boers & Driessen, "A particle-filter-based detection scheme", *IEEE Signal Process Letters*, vol. 10, no. 10, pp. 300–302, 2003.
- [17] S. Bonneau, M. Dahan, L. D. Cohen, "Single quantum dot tracking based on perceptual grouping using minimal paths in a spatiotemporal volume", *IEEE Transactions on Image Processing*, vol. 14, no. 9, pp. 1384–1395, 2005.
- [18] H. Bornfleth, P. Edelmann, D. Zink, T. Cremer, C. Cremer, "Quantitative motion analysis of subchromosomal foci in living cells using four-dimensional microscopy", *Biophysical Journal*, vol. 77, pp. 2871–2886, 1999.
- [19] H. Bornfleth, K. Satzler, R. Eils, C. Cremer, "High-precision distance measurements and volume-conserving segmentation of objects near and below the resolution limit in three-dimensional confocal fluorescence microscopy", *Journal of Microscopy*, vol. 189, no. 2, pp. 118–136, 1998.
- [20] E. J. Breen, G. H. Joss, K. L. Williams, "Locating objects of interest within biological images: The top hat box filter", *Journal of Computer-Assisted Microscopy*, vol. 3, no. 2, pp. 97–102, 1991.
- [21] D. S. Bright & E. B. Steel, "Two-dimensional top hat filter for extracting spots and spheres from digital images", *Journal of Microscopy*, vol. 146, no. 2, pp. 191–200, 1987.
- [22] J. C. Bulinski, D. J. Odde, B. J. Howell, T. D. Salmon, C. M. Waterman-Storer, "Rapid dynamics of the microtubule binding of ensconsin *in vivo*", *Journal of Cell Science*, vol. 114, no. 21, pp. 3885–3897, 2001.
- [23] P. Y. Burgi, A. L. Yuille, N. M. Grzywacz, "Probabilistic motion estimation based on temporal coherence", *Neural Computation*, vol. 12, no. 8, pp. 1839–1867, 2000.
- [24] M. B. Cannell, A. McMorland, C. Soeller, "Image enhancement by deconvolution", in *Handbook of Biological Confocal Microscopy*, B. Pawley (ed.), Springer, New York, Ch. 25, pp. 488–500, 2006.
- [25] J. F. Canny, "A computational approach to edge detection", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 8, no. 6, pp. 679–698, 1986.
- [26] B. C. Carter, G. T. Shubeita, S. P. Gross, "Tracking single particles: A user-friendly quantitative evaluation", *Physical Biology*, vol. 2, no. 1, pp. 60–72, 2005.
- [27] P. Carvalho, J. S. Tirnauer, D. Pellman, "Surfing on microtubule ends", *Trends in Cell Biology*, vol. 13, no. 5, pp. 229–237, 2003.
- [28] N. Caudron, I. Arnal, E. Buhler, D. Job, O. Valiron, "Microtubule nucleation from stable tubulin oligomers", *Journal of Biological Chemistry*, vol. 227, pp. 50973–50979, 2002.
- [29] D. P. Chakraborty, "Maximum likelihood analysis of free-response receiver operating characteristic (FROC) data", *Medical Physics*, vol. 16, no. 4, pp. 561–568, 1989.
- [30] D. P. Chakraborty, "Free-response methodology: Alternate analysis and a new observer-performance experiment", *Radiology*, vol. 174, pp. 873–881, 1990.
- [31] C. Chang, P. Ansari, A. Khokhar, "Efficient tracking of cyclic human motion by component motion", *IEEE Signal Processing Letters*, vol. 11, no. 12, pp. 941–944, 2004.
- [32] M. K. Cheezum, W. F. Walker, W. H. Guilford, "Quantitative comparison of algorithms for tracking single fluorescent particles", *Biophysical Journal*, vol. 81, no. 4, pp. 2378–2388, 2001.

- [33] D. Chetverikov & J. Verestói, “Feature point tracking for incomplete trajectories”, *Computing*, vol. 62, no. 4, pp. 321–338, 1999.
- [34] D. Comaniciu & P. Meer, “Mean shift: A robust approach toward feature space analysis”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 24, no. 5, pp. 603–619, 2002.
- [35] M. de Moraes Marim, Bo Zhang, J.-C. Olivo-Marin, C. Zimmer, “Improving single particle localization with an empirically calibrated Gaussian kernel”, in *Proceedings of the IEEE International Symposium on Biomedical Imaging*, pp. 1003–1006, 2008.
- [36] O. Debeir, P. Van Ham, R. Kiss, C. Decaestecker, “Tracking of migrating cells under phase-contrast video microscopy with combined mean-shift processes”, *IEEE Transactions on Medical Imaging*, vol. 24, no. 6, pp. 697–711, 2005.
- [37] A. Desai & T. J. Mitchison, “Microtubule polymerization dynamics”, *Annual Review of Cell and Developmental Biology*, vol. 13, pp. 83–117, 1997.
- [38] J. F. Dorn, G. Danuser, Y. Ge, “Computational processing and analysis of dynamic fluorescence image data”, *Methods in Cell Biology*, vol. 25, pp. 497–538, 2008.
- [39] A. Doucet, N. de Freitas, N. Gordon, *Sequential Monte Carlo Methods in Practice*, Springer-Verlag, Berlin, 2001.
- [40] A. Doucet, S. Godsill, C. Andrieu, “On sequential Monte Carlo sampling methods for Bayesian filtering”, *Statistics and Computing*, vol. 10, no. 3, pp. 197–208, 2000.
- [41] H. Driessen & Y. Boers, “An efficient particle filter for jump Markov nonlinear systems”, in *IEEE Target Tracking: Algorithms and Applications*, pp. 19–22, 2004.
- [42] D. A. Drubin, A. M. Garakani, P. A. Silver, “Motion as a phenotype: The use of live-cell imaging and machine visual screening to characterize transcription-dependent chromosome dynamics”, *BMC Cell Biology*, vol. 7:19, 2006.
- [43] A. Dufour, V. Shinin, S. Tajbakhsh, N. Guillen-Aghion adn J.-C. Olivo-Marin, C. Zimmer, “Segmenting and tracking fluorescent cells in dynamic 3-D microscopy with coupled active surfaces”, *IEEE Transactions on Image Processing*, vol. 14, no. 9, pp. 1396–1410, 2005.
- [44] O. Dzyubachyk, W. Niessen, E. Meijering, “Advanced level-set based multiple-cell segmentation and tracking in time-lapse fluorescence microscopy images”, in *IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pp. 185–188, 2008.
- [45] R. Eils & C. Athale, “Computational imaging in cell biology”, *Journal of Cell Biology*, vol. 161, no. 3, pp. 477–481, 2003.
- [46] P. Farla, R. Hersmus, J. Trapman, A. B. Houtsmuller, “Antiandrogens prevent stable DNA-binding of the androgen receptor”, *Journal of Cell Science*, vol. 118, no. 18, pp. 4187–4198, 2005.
- [47] M. A. T. Figueiredo & R. D. Nowak, “Wavelet-based image estimation: An empirical Bayes approach using Jeffrey’s noninformative prior”, *IEEE Transactions on Image Processing*, vol. 10, no. 9, pp. 1322–1331, 2001.
- [48] H. Flyvbjerg, T. E. Holy, S. Leibler, “Microtubule dynamics: Caps, catastrophes, and coupled hydrolysis”, *Physical Review E*, vol. 54, no. 5, pp. 5538–5560, 1996.
- [49] Y.S. Fong, C.A. Pomalaza, X.H. Wang, “Comparison study of nonlinear filters in image processing applications”, *Optical Engineering*, vol. 28, pp. 749–760, 1989.
- [50] Y. Freund & R. E. Schapire, “Experiments with a new boosting algorithm”, in *Proceedings of the 13th International Conference on Machine Learning*, pp. 148–156, 1996.
- [51] Y. Garini, B. J. Vermolen, I. T. Young, “From micro to nano: Recent advances in high-resolution microscopy”, *Current Opinion in Biotechnology*, vol. 16, no. 1, pp. 3–12, 2005.
- [52] A. Genovesio, T. Liedl, V. Emiliani, W. J. Parak, M. Coppey-Moisán, J.-C. Olivo-Marin, “Multiple particle tracking in 3-D+t microscopy: Method and application to the tracking of endocytosed quantum dots”, *IEEE Transactions on Image Processing*, vol. 15, no. 5, pp. 1062–1070, 2006.



- [53] D. Gerlich & J. Ellenberg, “4D imaging to assay complex dynamics in live specimens”, *Nature Cell Biology*, vol. 5, pp. S14–S19, 2003.
- [54] D. Gerlich, J. Mattes, R. Eils, “Quantitative motion analysis and visualization of cellular structures”, *Methods*, vol. 29, no. 1, pp. 3–13, 2003.
- [55] S. F. Gibson & F. Lanni, “Experimental test of an analytical model of aberration in an oil-immersion objective lens used in three-dimensional light microscopy”, *Journal of the Optical Society of America A: Optics and Image Science*, vol. 9, no. 1, pp. 154–166, 1992.
- [56] S. Godsill & J. Vermaak, “Variable rate particle filters for tracking applications”, in *Proceedings of the IEEE/SP 13th Workshop on Statistical Signal Processing*, pp. 1280–1285, 2005.
- [57] S.J. Godsill, J. Vermaak, K-F. Ng, J-F. Li, “Models and algorithms for tracking of manoeuvring objects using variable rate particle filters”, *Proceedings of the IEEE*, vol. 95, no. 5, pp. 925–952, 2007.
- [58] I. Grigoriev, D. Splinter, N. Keijzer, P. S. Wulf, J. Demmers, T. Ohtsuka, M. Modesti, I. V. Maly, F. Grosveld, C. C. Hoogenraad, A. Akhmanova, “Rab6 regulates transport and targeting of exocytotic carriers”, *Developmental Cell*, vol. 13, no. 2, pp. 305–314, 2007.
- [59] M. Gu, *Advanced Optical Imaging Theory*, Springer-Verlag, Berlin, 2000.
- [60] M. G. L. Gustafsson, “Extended resolution fluorescence microscopy”, *Current Opinion in Structural Biology*, vol. 9, no. 5, pp. 627–634, 1999.
- [61] N. M. Haan & S. J. Godsill, “A time-varying model for DNA Sequencing data submerged in correlated noise”, in *Proceedings of the IEEE Workshop on Statistical Signal Processing*, August 2001.
- [62] S. Hadjidemetriou, J. S. Duncan, D. Toomre, D. Tuck, “Automatic quantification of microtubule dynamics”, in *Proceedings of the IEEE International Symposium on Biomedical Imaging*, pp. 656–659, 2004.
- [63] S. W. Hell, M. Dyba, S. Jakobs, “Concepts for nanoscale resolution in fluorescence microscopy”, *Current Opinion in Neurobiology*, vol. 14, no. 5, pp. 599–609, 2004.
- [64] B. Hinz, W. Alt, C. Johnen, V. Herzog, H.-W. Kaiser, “Quantifying lamella dynamics of cultured cells by SACED, a new computer-assisted motion analysis”, *Experimental Cell Research*, vol. 251, no. 1, pp. 234–243, 1999.
- [65] T. J. Holmes, D. Biggs, A. Abu-Tarif, “Blind deconvolution”, in *Handbook of Biological Confocal Microscopy*, B. Pawley (ed.), Springer, New York, Ch. 24, pp. 468–487, 2006.
- [66] B. K. Horn, *Robot Vision*, MIT Press, 1986.
- [67] J. Howard & A. A. Hyman, “Dynamics and mechanics of the microtubule plus end”, *Nature*, vol. 422, no. 6933, pp. 753–758, 2003.
- [68] C. Hue, J.-P. Le Cadre, P.F. Perez, “Sequential Monte Carlo methods for multiple target tracking and data fusion”, *IEEE Transactions on Signal Processing*, vol. 50, no. 2, pp. 309–325, 2002.
- [69] S. Inoue, *Handbook of Optics*, McGraw-Hill, Inc., New York, 1995.
- [70] M. Isard & A. Blake, “CONDENSATION – Conditional density propagation for visual tracking”, *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5–28, 1998.
- [71] K. Jaqaman, D. Loerke, M. Mettlen, H. Kuwata, S. Grinstein, S.L. Schmid, G. Danuser, “Robust single-particle tracking in live-cell time-lapse sequences”, *Nature Methods*, vol. 5, no. 8, pp. 695–702, 2008.
- [72] K. Jaqaman, D. Loerke, M. Mettlen, H. Kuwata, S. Grinstein, S. L. Schmid, G. Danuser, “Robust single-particle tracking in live-cell time-lapse sequences”, *Nature Methods*, vol. 5, no. 8, pp. 695–702, 2008.
- [73] S. Jiang, X. Zhou, T. Kirchhausen, S. T. C. Wong, “Detection of molecular particles in live cells via machine learning”, *Cytometry Part A*, vol. 71, no. 8, pp. 563–575, 2007.
- [74] S. Jiang, X. Zhou, T. Kirchhausen, S. T. C. Wong, “Tracking molecular particles in live cells using fuzzy rule-based system”, *Cytometry Part A*, vol. 71, no. 8, pp. 576–584, 2007.

- [75] Y. Kalaidzidis, "Intracellular objects tracking", *European Journal of Cell Biology*, vol. 86, no. 9, pp. 569–578, 2007.
- [76] Z. Khan, T. Balch, F. Dellaert, "MCMC-based particle filtering for tracking a variable number of interacting targets", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 11, pp. 1805–1819, 2005.
- [77] M. Klaas, N. de Freitas, A. Doucet, "Toward Practical  $N^2$  Monte Carlo: The Marginal Particle Filter", in *Proceedings of the Annual Conference on Uncertainty in Artificial Intelligence*, pp. 308–331, 2005.
- [78] Y. Komarova, C. O. de Groot, I. Grigoriev, S. Montenegro Gouveia, E. L. Munteanu, J. M. Schober, S. Honnappa, R. M. Buey, C. C. Hoogenraad, M. Dogterom, G. G. Borisy, M. O. Steinmetz, A. Akhmanova, "Mammalian end binding proteins control persistent microtubule growth", *Journal of Cell Biology*, vol. 184, no. 5, pp. 691–706, 2009.
- [79] Y. Kosuge & T. Matsuzaki, "The optimum gate shape and threshold for target tracking", in *Proceedings of the SICE Annual Conference in Fukui*, vol. 2, pp. 2152–2157, 2003.
- [80] G. Lansbergen & A. Akhmanova, "Microtubule plus end: A hub of cellular activities", *Traffic*, vol. 7, no. 5, pp. 499–507, 2006.
- [81] G. Lansbergen, I. Grigoriev, Y. Mimori-Kiyosue, T. Ohtsuka, S. Higa, I. Kitajima, J. Demmers, N. Galjart, A. B. Houtsmuller, F. Grosveld, A. Akhmanova, "CLASPs attach microtubule plus ends to the cell cortex through a complex with LL5 $\beta$ ", *Developmental Cell*, vol. 11, no. 1, pp. 21–32, 2006.
- [82] V. Levi, Q. Ruan, E. Gratton, "3-D particle tracking in a two-photon microscope: Application to the study of molecular dynamics in cells", *Biophysical Journal*, vol. 88, no. 4, pp. 2919–2928, 2005.
- [83] K. Li, E. Miller, L. Weiss, P. Campbell, T. Kanade, "Online tracking of migrating and proliferating cells imaged with phase-contrast microscopy", in *Proceedings of the Conference on Computer Vision and Pattern Recognition Workshop*, pp. 65–72, 2006.
- [84] X. R. Li & V. P. Jilkov, "Survey of maneuvering target tracking: Part I: Dynamic models", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1333–1364, 2003.
- [85] R. Lienhart & J. Maydt, "An extended set of Haar-like features for rapid object detection", in *Proceedings of the 2002 International Conference on Image Processing*, vol. 1, pp. I–900–I–903, 2002.
- [86] T. Lindeberg, "Feature detection with automatic scale selection", *International Journal of Computer Vision*, vol. 30, no. 2, pp. 79–116, 1998.
- [87] J. Lippincott-Schwartz, N. Altan-Bonnet, G. H. Patterson, "Photobleaching and photoactivation: Following protein dynamics in living cells", *Nature Cell Biology*, vol. 5, pp. S7–S13, 2003.
- [88] J. Lippincott-Schwartz & G. H. Patterson, "Development and use of fluorescent protein markers in living cells", *Science*, vol. 300, no. 5616, pp. 87–91, 2003.
- [89] J. MacCormick & A. Blake, "Probabilistic exclusion and partitioned sampling for multiple object tracking", *International Journal of Computer Vision*, vol. 39, no. 1, pp. 57–71, 2000.
- [90] R. P. S. Mahler, "Multitarget Bayes filtering via first-order multitarget moments", *IEEE Transactions on Aerospace and Electronic Systems*, vol. 39, no. 4, pp. 1152–1178, 2003.
- [91] S. Mallat, *A Wavelet Tour of Signal Processing*, Academic Press, 1998.
- [92] S. B. Marston, I. D. C. Fraser, W. Bing, G. Roper, "A simple method for automatic tracking of actin filaments in the motility assay", *Journal of Muscle Research and Cell Motility*, vol. 17, no. 4, pp. 497–506, 1996.
- [93] G. J. McLachlan, *Discriminant Analysis and Statistical Pattern Recognition*, Wiley-Interscience, 2004.
- [94] E. Meijering, "MTrackJ: A Java program for manual object tracking", <http://www.imagescience.org/meijering/software/mtrackj/>, 2006.

- [95] E. Meijering, I. Smal, G. Danuser, "Tracking in molecular bioimaging", *IEEE Signal Processing Magazine*, vol. 23, no. 3, pp. 46–53, 2006.
- [96] E. Meijering, I. Smal, O. Dzyubachyk, J.-C. Olivo-Marin, "Time-lapse imaging", in *Microscope Image Processing*, Q. Wu, F. A. Merchant, K. R. Castleman (eds.), Elsevier Academic Press, Burlington, MA, pp. 401–440, 2008.
- [97] E. Meijering & G. van Cappellen, "Quantitative biological image analysis", in *Imaging Cellular and Molecular Biological Functions*, S. L. Shorte & F. Frischknecht (eds.), Springer-Verlag, Berlin, pp. 45–70, 2007.
- [98] Y. Mimori-Kiyosue, I. Grigoriev, G. Lansbergen, H. Sasaki, C. Matsui, F. Severin, N. Galjart, F. Grosveld, I. Vorobjev, S. Tsukita, A. Akhmanova, "CLASP1 and CLASP2 bind to EB1 and regulate microtubule plus-end dynamics at the cell cortex", *Journal of Cell Biology*, vol. 168, no. 1, pp. 141–153, 2005.
- [99] T. Mitchison & M. Kirschner, "Dynamic instability of microtubule growth", *Nature*, vol. 312, no. 5991, pp. 237–242, 1984.
- [100] A. Miyawaki, A. Sawano, T. Kogure, "Lighting up cells: Labelling proteins with fluorophores", *Nature Cell Biology*, vol. 5, pp. S1–S7, 2003.
- [101] E. L. Munteanu, *Dynamics and regulation at the tip: A high resolution view on microtubule assembly*, Ph.D. thesis, Institute for Atomic and Molecular Physics (AMOLF), Amsterdam, 2008.
- [102] D. Murphy, *Fundamentals of Light Microscopy and Digital Imaging*, Wiley-Liss, New York, 2001.
- [103] B. Neumann, M. Held, U. Liebel, H. Erfle, P. Rogers, R. Pepperkok, J. Ellenberg, "High-throughput RNAi screening by time-lapse imaging of live human cells", *Nature Methods*, vol. 3, no. 5, pp. 385–390, 2006.
- [104] W. Ng, J. Li, S. Godsill, J. Vermaak, "A hybrid method for online joint detection and tracking for multiple targets", *IEEE Transactions on Aerospace and Electronic Systems*, in press.
- [105] W. Ng, J. Li, S. Godsill, J. Vermaak, "A hybrid approach for online joint detection and tracking for multiple targets", in *Proceedings of the 2005 IEEE Aerospace Conference*, pp. 2126–2141, 2005.
- [106] P. Niethammer, I. Kronja, S. Kandels-Lewis, S. Rybina, P. Bastiaens, E. Karsenti, "Discrete states of a protein interaction network govern interphase and mitotic microtubule dynamics", *PLoS Biology*, vol. 5, no. 2, p. e29, 2007.
- [107] E. Nogales, M. Whittaker, R.A. Milligan, K.H. Downing, "High-resolution model of the microtubule", *Cell*, vol. 96, pp. 79–88, 1999.
- [108] J.-C. Olivo-Marin, "Extraction of spots in biological images using multiscale products", *Pattern Recognition*, vol. 35, no. 9, pp. 1989–1996, 2002.
- [109] P. Pankajakshan, Bo Zhang, L. Blanc-Feraud, Z. Kam, J.-C. Olivo-Marin, J. Zerubia, "Blind deconvolution for diffraction-limited fluorescence microscopy", in *Proceedings of the IEEE International Symposium on Biomedical Imaging*, pp. 740–743, 2008.
- [110] M. Pantic & I. Patras, "Dynamics of facial expression: Recognition of facial actions and their temporal segments from face profile image sequences", *IEEE Transactions on Systems, Man, and Cybernetics, Part B*, vol. 36, no. 2, pp. 433–449, 2006.
- [111] C.P. Papageorgiou, M. Oren, T. Poggio, "A general framework for object detection", in *Proceedings of the 6th International Conference on Computer Vision*, pp. 555–562, 1998.
- [112] J. B. Pawley, *Handbook of Biological Confocal Microscopy*, 3rd ed., Springer, New York, 2006.
- [113] R. Pepperkok & J. Ellenberg, "High-throughput fluorescence microscopy for systems biology", *Nature Reviews Molecular Cell Biology*, vol. 7, no. 9, pp. 690–696, 2006.
- [114] F. Perez, G. S. Diamantopoulos, R. Stalder, T. E. Kreis, "CLIP-170 highlights growing microtubule ends in vivo", *Cell*, vol. 96, no. 4, pp. 517–527, 1999.

- [115] P. Perez, J. Vermaak, A. Blake, "Data fusion for visual tracking with particles", *Proceedings of the IEEE*, vol. 92, no. 3, pp. 495–513, 2004.
- [116] P. Perona & J. Malik, "Scale-space and edge detection using anisotropic diffusion", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 7, pp. 629–639, 1990.
- [117] M. Piccardi, "Background subtraction techniques: A review", in *2004 IEEE International Conference on Systems, Man and Cybernetics*, vol. 4, pp. 3099–3104, 2004.
- [118] X. Qu, D. Wu, L. Mets, N. F. Scherer, "Nanometer-localized multiple single-molecule fluorescence microscopy", *PNAS*, vol. 101, no. 31, pp. 11298–11303, 2004.
- [119] G. Rabut & J. Ellenberg, "Automatic real-time three-dimensional cell tracking by fluorescence microscopy", *Journal of Microscopy*, vol. 216, no. 2, pp. 131–137, 2004.
- [120] T. Ragan, H. Huang, P. So, E. Gratton, "3D particle tracking on a two-photon microscope", *Journal of Fluorescence*, vol. 16, no. 3, pp. 325–336, 2006.
- [121] W. S. Rasband, "ImageJ: Image processing and analysis in Java", <http://rsb.info.nih.gov/ij/>.
- [122] N. Ray & S. T. Acton, "Motion gradient vector flow: An external force for tracking rolling leukocytes with shape and size constrained active contours", *IEEE Transactions on Medical Imaging*, vol. 23, no. 12, pp. 1466–1478, 2004.
- [123] B. Rieger, C. Molenaar, R. W. Dirks, L. J. van Vliet, "Alignment of the cell nucleus from labeled proteins only for 4D in vivo imaging", *Microscopy Research and Technique*, vol. 64, no. 2, pp. 142–150, 2004.
- [124] J. P. Rigaut & J. Vassy, "High-resolution three-dimensional images from confocal scanning laser microscopy. Quantitative study and mathematical correction of the effects from bleaching and fluorescence attenuation in depth", *Analytical and Quantitative Cytology and Histology*, vol. 13, no. 4, pp. 223–232, 1991.
- [125] B. D. Ripley, *Stochastic Simulation*, Wiley, New York, 1987.
- [126] B. Ristic, S. Arulampalam, N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*, Artech House, Boston, 2004.
- [127] M. Saban, A. Altinok, A. Peck, C. Kenney, S. Feinstein, L. Wilson, K. Rose, B. S. Manjunath, "Automated tracking and modeling of microtubule dynamics", in *Proceedings of the IEEE International Symposium on Biomedical Imaging*, pp. 1032–1035, 2006.
- [128] D. Sage, F. R. Neumann, F. Hediger, S. M. Gasser, M. Unser, "Automatic tracking of individual fluorescence particles: Application to the study of chromosome dynamics", *IEEE Transactions on Image Processing*, vol. 14, no. 9, pp. 1372–1383, 2005.
- [129] P. Sarder & A. Nehorai, "Deconvolution Methods for 3-D Fluorescence Microscopy Images", *IEEE Signal Processing Magazine*, vol. 23, no. 3, pp. 32–45, May 2006.
- [130] Y. Sato, J. Chen, R. A. Zoroofi, N. Harada, S. Tamura, T. Shiga, "Automatic extraction and measurement of leukocyte motion in microvessels using spatiotemporal image analysis", *IEEE Transactions on Biomedical Engineering*, vol. 44, pp. 225–236, 1997.
- [131] M. J. Saxton & K. Jacobson, "Single-particle tracking: Applications to membrane dynamics", *Annual Review of Biophysics and Biomolecular Structure*, vol. 26, pp. 373–399, June 1997.
- [132] I. F. Sbalzarini & P. Koumoutsakos, "Feature point tracking and trajectory analysis for video imaging in cell biology", *Journal of Structural Biology*, vol. 151, no. 2, pp. 182–195, 2005.
- [133] M. Schaap, R. Manniesing, I. Smal, T. van Walsum, A. van der Lugt, W. J. Niessen, "Bayesian tracking of tubular structures and its application to carotid arteries in CTA", in *Proceedings of Medical Image Computing and Computer-Assisted Intervention*, pp. 562–570, 2007.
- [134] M. Schaap, I. Smal, C. T. Metz, T. van Walsum, W. J. Niessen, "Bayesian tracking of elongated structures in 3D images", in *Proceedings of Information Processing in Medical Imaging, 20th International Conference*, pp. 74–85, 2007.
- [135] D. A. Schiffmann, D. Dikovskaya, P. L. Appleton, I. P. Newton, D. A. Creager, C. Allan, I. S. Näthke, I. G. Goldberg, "Open microscopy environment and FindSpots: Integrating image informatics with quantitative multidimensional image analysis", *BioTechniques*, vol. 41, no. 2, pp. 199–208, 2006.

- [136] W. Schroeder, K. Martin, B. Lorensen, *The Visualization Toolkit: An Object-Oriented Approach to 3D Graphics*, 3rd ed., Kitware, New York, 2002.
- [137] S. C. Schuyler & D. Pellman, "Microtubule "plus-end-tracking proteins": The end is just the beginning", *Cell*, vol. 105, no. 4, pp. 421–424, 2001.
- [138] J. Serra, *Image Analysis and Mathematical Morphology*, Academic Press, London, 1982.
- [139] H. Shen, G. Nelson, S. Kennedy, D. Nelson, J. Johnson, D. Spiller, M. R. H. White, D. B. Kell, "Automatic tracking of biological cells and compartments using particle filters and active contours", *Chemometrics and Intelligent Laboratory Systems*, vol. 82, no. 1-2, pp. 276–282, 2006.
- [140] I. Smal, K. Draegestein, N. Galjart, W. Niessen, E. Meijering, "Rao-Blackwellized marginal particle filtering for multiple object tracking in molecular bioimaging", in *Proceedings of the International Conference on Information Processing in Medical Imaging*, pp. 110–121, 2007.
- [141] I. Smal, K. Draegestein, N. Galjart, W. Niessen, E. Meijering, "Particle filtering for multiple object tracking in dynamic fluorescence microscopy images: Application to microtubule growth analysis", *IEEE Transactions on Medical Imaging*, vol. 27, no. 6, pp. 789–804, 2008.
- [142] 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.
- [143] I. Smal, W. Niessen, E. Meijering, "Bayesian tracking for fluorescence microscopic imaging", in *Proceedings of the IEEE International Symposium on Biomedical Imaging*, pp. 550–553, 2006.
- [144] I. Smal, W. Niessen, E. Meijering, "Particle filtering for multiple object tracking in molecular cell biology", in *Proceedings of the Nonlinear Statistical Signal Processing Workshop*, pp. 44.1–44.4, 2006.
- [145] I. Smal, W. Niessen, E. Meijering, "Advanced particle filtering for multiple object tracking in dynamic fluorescence microscopy images", in *Proceedings of the IEEE International Symposium on Biomedical Imaging*, pp. 1048–1051, 2007.
- [146] I. Smal, W. Niessen, E. Meijering, "A new detection scheme for multiple object tracking in fluorescence microscopy by joint probabilistic data association filtering", in *Proceedings of the IEEE International Symposium on Biomedical Imaging*, pp. 264–267, 2008.
- [147] P. Soille, *Morphological Image Analysis: Principles and Applications*, Springer-Verlag, Berlin, 2003.
- [148] L. Song, E. J. Hennink, I. T. Young, H. J. Tanke, "Photobleaching kinetics of fluorescein in quantitative fluorescence microscopy", *Biophysical Journal*, vol. 68, no. 6, pp. 2588–2600, 1995.
- [149] M. Sonka, V. Hlavac, R. Boyle, *Image Processing, Analysis, and Machine Vision*, 2nd ed., PWS Publishing, Pacific Grove, 1999.
- [150] C. O. S. Sorzano, P. Thévenaz, M. Unser, "Elastic registration of biological images using vector-spline regularization", *IEEE Transactions on Biomedical Engineering*, vol. 52, no. 4, pp. 652–663, 2005.
- [151] J.-L. Starck, M. Elad, D.L. Donoho, "Redundant multiscale transforms and their application for morphological component analysis", in *Advances in Imaging and Electron Physics*, vol. 132, Elsevier Academic Press, San Diego, CA, pp. 288–348, 2004.
- [152] J.-L. Starck, J. Fadili, F. Murtagh, "The undecimated wavelet decomposition and its reconstruction", *IEEE Transactions on Image Processing*, vol. 16, no. 2, pp. 297–309, 2007.
- [153] J.-L. Starck & F. Murtagh, "Astronomical image and signal processing: looking at noise, information and scale", *IEEE Signal Processing Magazine*, vol. 18, no. 2, pp. 30–40, 2001.
- [154] J.-L. Starck & F. Murtagh, *Astronomical Image and Data Analysis*, Springer-Verlag, Berlin, 2002.

- [155] T. Stepanova, J. Slemmer, C. C. Hoogenraad, G. Lansbergen, B. Dortland, C. I. de Zeeuw, F. Grosveld, G. van Cappellen, A. Akhmanova, N. Galjart, "Visualization of microtubule growth in cultured neurons via the use of EB3-GFP (end-binding protein 3-green fluorescent protein)", *Journal of Neuroscience*, vol. 23, no. 7, pp. 2655–2664, 2003.
- [156] D. J. Stephens & V. J. Allan, "Light microscopy techniques for live cell imaging", *Science*, vol. 300, no. 5616, pp. 82–86, 2003.
- [157] S. R. Sternberg, "Biomedical image processing", *Computer*, vol. 16, no. 1, pp. 22–34, 1983.
- [158] J. Suh, M. Dawson, J. Hanes, "Real-time multiple-particle tracking: Application to drug and gene delivery", *Advanced Drug Delivery Reviews*, vol. 57, no. 1, pp. 63–78, 2005.
- [159] B. M. ter Haar Romeny, *Front-End Vision and Multi-Scale Image Analysis*, Springer, Berlin, 2003.
- [160] D. Thomann, J. Dorn, P. K. Sorger, G. Danuser, "Automatic fluorescent tag localization II: Improvement in super-resolution by relative tracking", *Journal of Microscopy*, vol. 211, no. 3, pp. 230–248, 2003.
- [161] D. Thomann, D. R. Rines, P. K. Sorger, G. Danuser, "Automatic fluorescent tag detection in 3D with super-resolution: Application to the analysis of chromosome movement", *Journal of Microscopy*, vol. 208, no. 1, pp. 49–64, 2002.
- [162] R. E. Thompson, D. R. Larsonand, W. W. Webb, "Precise nanometer localization analysis for individual fluorescent probes", *Biophysical Journal*, vol. 82, pp. 2775–2783, 2002.
- [163] C. Tomasi & R. Manduchi, "Bilateral filtering for gray and color images", in *Proceedings of the Sixth International Conference on Computer Vision*, pp. 839–846, 1998.
- [164] R. Y. Tsien, "Imaging imaging's future", *Nature Cell Biology*, vol. 5, pp. S16–S21, 2003.
- [165] G. L. Turin, "An introduction to matched filters", *IRE Transactions on Information Theory*, vol. 6, no. 3, pp. 311–329, 1960.
- [166] W. Tvaruskó, M. Bentele, T. Misteli, R. Rudolf, C. Kaether, D. L. Spector, H. H. Gerdes, R. Eils, "Time-resolved analysis and visualization of dynamic processes in living cells", *Proceedings of the National Academy of Sciences of the United States of America*, vol. 96, no. 14, pp. 7950–7955, 1999.
- [167] D. Uttenweiler, C. Veigel, R. Steubing, C. Götz, S. Mann, H. Haussecker, B. Jähne, R. H. A. Fink, "Motion determination in actin filament fluorescence images with a spatio-temporal orientation analysis method", *Biophysical Journal*, vol. 78, no. 5, pp. 2709–2715, 2000.
- [168] D. Uttenweiler, C. Weber, B. Jähne, R. H. Fink, H. Scharr, "Spatiotemporal anisotropic diffusion filtering to improve signal-to-noise ratios and object restoration in fluorescence microscopic image sequences", *Journal of Biomedical Optics*, vol. 8, no. 1, pp. 40–47, 2003.
- [169] P. Vallotton, A. Ponti, C. M. Waterman-Storer, E. D. Salmon, G. Danuser, "Recovery, visualization, and analysis of actin and tubulin polymer flow in live cells: A fluorescent speckle microscopy study", *Biophysical Journal*, vol. 85, no. 2, pp. 1289–1306, 2003.
- [170] M. E. Van Royen, S. M. Cunha, M. C. Brink, K. A. Mattern, A. L. Nigg, H. J. Dubbink, P. J. Verschure, J. Trapman, A. B. Houtsmuller, "Compartmentalization of androgen receptor protein-protein interactions in living cells", *Journal of Cell Biology*, vol. 177, no. 1, pp. 63–72, 2007.
- [171] C. J. Veenman, M. J. T. Reinders, E. Backer, "Motion tracking as a constrained optimization problem", *Pattern Recognition*, vol. 36, no. 9, pp. 2049–2067, 2003.
- [172] F. Verde, M. Dogterom, E. Stelzer, E. Karsenti, S. Leibler, "Control of microtubule dynamics and length by cyclin A- and cyclin B-dependent kinases in *Xenopus* egg extracts", *Journal of Cell Biology*, vol. 118, no. 5, pp. 1097–1108, 1992.
- [173] F. Verde, J. C. Labbe, M. Doree, E. Karsenti, "Regulation of microtubule dynamics by cdc2 protein kinase in cell-free extracts of *Xenopus* eggs", *Nature*, vol. 343, no. 6255, pp. 233–238, 1990.
- [174] J. Vermaak, A. Doucet, P. Pérez, "Maintaining multi-modality through mixture tracking", in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1110–1116, 2003.



- [175] J. Vermaak, N. Ikoma, S. J. Godsill, "Extended object tracking using particle techniques", in *Proceedings of the IEEE Aerospace Conference*, vol. 3, pp. 1876–1885, 2004.
- [176] B. J. Vermolen, Y. Garini, I. T. Young, R. W. Dirks, V. Raz, "Segmentation and analysis of the three-dimensional redistribution of nuclear components in human mesenchymal stem cells", *Cytometry Part A*, vol. 73, no. 9, pp. 816–824, 2008.
- [177] L. Vincent, "Morphological grayscale reconstruction in image analysis: Applications and efficient algorithms", *IEEE Transactions on Image Processing*, vol. 2, no. 2, pp. 176–201, 1993.
- [178] P. Viola & M. Jones, "Rapid object detection using a boosted cascade of simple features", in *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, pp. I-511–I-518, 2001.
- [179] L.J. Van Vliet, D. Sudar, I. T. Young, "Digital fluorescence imaging using cooled charge-coupled device array cameras", in *Cell Biology: a Laboratory Handbook*, K. Simons (ed.), 2nd. ed., Academic Press, New York, pp. 109–120, 1998.
- [180] C. Vonesch, F. Aguet, J.-L. Vonesch, M. Unser, "The colored revolution of bioimaging", *IEEE Signal Processing Magazine*, vol. 23, no. 3, pp. 20–31, 2006.
- [181] T. Voorn-Brouwer, A. Kragt, H.F. Tabak, B. Distel, "Peroxisomal membrane proteins are properly targeted to peroxisomes in the absence of COPI- and COPII-mediated vesicular transport", *Journal of Cell Science*, vol. 114, no. 11, pp. 2199–2204, 2001.
- [182] R.A. Walker, E.T. O'Brien, N.K. Pryer, M.F. Soboeiro, W.A. Voter, H.P. Erickson, E.D. Salmo, "Dynamic instability of individual microtubules analyzed by video light microscopy: rate constants and transition frequencies", *Journal of Cell Biology*, vol. 107, pp. 1437–1448, 1988.
- [183] Q. Wen, J. Gao, A. Kosaka, H. Iwaki, K. Luby-Phelps, D. Mundy, "A particle filter framework using optimal importance function for protein molecules tracking", in *Proceedings of the IEEE International Conference on Image Processing*, vol. 1, pp. 1161–1164, 2005.
- [184] J. Wolf, W. Burgard, H. Burkhardt, "Robust vision-based localization by combining an image-retrieval system with Monte Carlo localization", *IEEE Transactions on Robotics*, vol. 21, no. 2, pp. 208–216, 2005.
- [185] Q. Wu, F. A. Merchant, K. R. Castleman, *Microscope Image Processing*, Elsevier Academic Press, Burlington, MA, 2008.
- [186] Y. Wu, J. Lin, T. S. Huang, "Analyzing and capturing articulated hand motion in image sequences", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 12, pp. 1910–1922, 2005.
- [187] K. Yin Kong, A. I. Marcus, J. Young Hong, P. Giannakakou, M. D. Wang, "Computer assisted analysis of microtubule dynamics in living cells", in *Proceedings of the 27th Annual International Conference of the IEEE Engineering in Medicine and Biology Society*, pp. 3982–3985, 2005.
- [188] I. Zaliapin, A. Gabrielov, V. Keilis-Borok, "Multiscale trend analysis", *Fractals*, vol. 12, no. 3, pp. 275–292, 2004.
- [189] I. Zaliapin, I. Semenova, A. Kashina, V. Rodionov, "Multiscale trend analysis of microtubule transport in melanophores", *Biophysical Journal*, vol. 88, pp. 4008–4016, 2005.
- [190] B. Zhang, J. Zerubia, J.-C. Olivo-Marin, "Gaussian approximations of fluorescence microscope point-spread function models", *Applied Optics*, vol. 46, no. 10, pp. 1819–1829, 2007.
- [191] X. Zhou & S. T. C. Wong, "Informatics challenges of high-throughput microscopy", *IEEE Signal Processing Magazine*, vol. 23, no. 3, pp. 63–72, 2006.
- [192] C. Zimmer, E. Labruyère, V. Meas-Yedid, N. Guillén, J.-C. Olivo-Marin, "Segmentation and tracking of migrating cells in videomicroscopy with parametric active contours: a tool for cell-based drug testing", *IEEE Transactions on Medical Imaging*, vol. 21, no. 10, pp. 1212–1221, 2002.



---

---

## Samenvatting

---

De ontwikkeling van fluorescentie microscopie heeft in de afgelopen tien jaar bijgedragen aan revolutionaire vooruitgangen in de biologie en heeft nieuwe wegen geopend voor het bestuderen van intracellulaire dynamische processen. Een eenvoudige beschrijving van soms complexe bewegingspatronen in levende cellen kan inzicht geven in de onderliggende mechanismen die deze bewegingen bepalen. Nauwkeurige en reproduceerbare methoden voor het verwerken en analyseren van de beelden die gemaakt worden voor dergelijke studies zijn echter nog schaars. Omdat handmatige beeldanalyse tijdrovend is, mogelijk onnauwkeurig, en slecht reproduceerbaar, worden veel vraagstukken die biologisch zeer relevant zijn niet behandeld, of beantwoord met grote onzekerheid. Het is daarom cruciaal dat er automatische beeldanalysetechnieken worden ontwikkeld voor het accuraat en reproduceerbaar volgen en analyseren van intracellulaire structuren in microscopische beeldseries.

Recente onderzoeken hebben aangetoond dat het menselijk visueel systeem in hoge mate spatiële en temporele informatie integreert. Locale bewegingssignalen zijn vaak voor velerlei uitleg vatbaar, en veel van deze dubbelzinnige verschijnselen kunnen worden opgelost door het veronderstellen van samenhang in de tijd. Men heeft daarom voorgesteld om invoerdata te groeperen in de tijd en deze daarna te gebruiken voor het voorspellen en schatten van de bewegingsstromen in beeldseries. Een dergelijke groepering in de tijd kan worden beschreven in termen van een Bayesiaanse generalisatie van het zogenaamde Kalman filter. Bestaande technieken, hetzij commercieel, hetzij academisch, voor het volgen van objecten in beeldseries maken in het algemeen zeer beperkt gebruik van tijdsinformatie en a priori kennis over de (beweging van de) objecten.

Het thema van deze dissertatie is de ontwikkeling en toepassing van zogenaamde “particle filtering” methoden voor het volgen van meerdere subcellulaire objecten in biologische beeldseries. Particle filtering (PF) is een techniek voor het uitvoeren van recursieve Bayesiaanse schattingen door middel van Monte Carlo bemonstering. De Bayesiaanse aanpak biedt de mogelijkheid om informatie over het beeldvormende systeem, mogelijke ruisbronnen, en het dynamisch gedrag van de te volgen objecten te modelleren in termen van kansdichtheden. Het is echter meer een recept dan een direct toepasbare oplossing voor een gegeven probleem, en moet daarom in de praktijk geconcretiseerd en geïmplementeerd worden, bijvoorbeeld met behulp van de PF benadering. De keuze voor een PF benadering levert echter niet een uniek algoritme op, en er zullen voor verschillende toepassingen verschillende algoritmische afwegingen moeten worden gemaakt. Zoals door sommige methodologen in het vakgebied wordt

gezegd, is het ontwerpen van efficiënte PF algoritmen nog steeds meer een kunst dan een wetenschap. In deze dissertatie wordt een aantal nieuwe PF methoden beschreven voor intracellulaire bewegingsanalyse. De toepasbaarheid van deze methoden wordt gedemonstreerd en geëvalueerd, zowel voor robuuste en nauwkeurige detectie en het volgen van grote aantallen kleine objecten in 2D en 3D beeldseries verkregen met behulp van fluorescentie microscopie, als voor de analyse van dynamische parameters in zogenaamde kymografische beelden.

In tegenstelling tot de conventionele twee-staps methoden (spatiële detectie gevolgd door temporele associatie) voor het volgen van objecten, heeft de Bayesiaanse aanpak geen afzonderlijke object-detectie procedure nodig. Robuuste en nauwkeurige object-detectoren komen echter wel degelijk van pas in het Bayesiaanse kader. In het bijzonder kunnen ze worden gebruikt in beslissingsprocedures voor het starten van nieuwe objectpaden en het beëindigen van de bestaande paden, respectievelijk wanneer een nieuw object verschijnt of een bestaand object uit het blikveld verdwijnt. In Hoofdstuk 2 worden acht methoden voor de detectie van kleine objecten in fluorescentie-microscopiebeelden kwantitatief geëvalueerd. Er wordt aangetoond dat zogenaamde “machine-learning” (ML) methoden (in dit geval AdaBoost en Fisher discriminant-analyse) in het algemeen beter presteren. Deze methoden resulteren in de hoogste correct-positieve detectie ratio (bij een zeer lage fout-positieve detectie ratio) en de kleinste gevoeligheid voor parameterwijzigingen, voor alle beschouwde beelddata. De verschillen in prestatie tussen de ML en sommige van de overige methoden zijn echter niet groot, vooral niet als wordt vergeleken met de zogenaamde h-dome detector (HD), die in het hoofdstuk wordt voorgesteld. Op basis van de resultaten wordt de conclusie getrokken dat wanneer een detector met goede algemene prestaties is vereist, de genoemde ML methoden of de HD detector de voorkeur verdienen. Het nadeel van de eerstgenoemde methoden is echter dat ze afhankelijk zijn van een trainingsfase, waarvoor eerst positieve en negatieve monsters uit de beelddata dienen te worden geëxtraheerd. Dit vereist de handmatige annotatie van duizenden objecten om voldoende onderscheidingsvermogen te verkrijgen, wat niet alleen vervelend en tijdrovend werk is, maar bovendien waarnemer-afhankelijke resultaten oplevert. Met het oog hierop is de HD detector veel gemakkelijker in het gebruik. Wel moet worden opgemerkt dat wanneer de signaal-ruisverhouding voldoende hoog is ( $>5$  als vuistregel), alle bestudeerde methoden vergelijkbaar presteren, en er slechts minimale aanpassing van de parameters nodig is voor gebruik in een specifieke toepassing.

Hoofdstuk 3 beschrijft het ontwerp van een nieuw PF algoritme voor kwantitatieve analyse van de beweging van subcellulaire objecten, in dit geval de groei van microtubuli. Het algoritme gebruikt a priori informatie over het dynamisch gedrag van microtubuli en het beeldingvormingsproces. Hierdoor levert het bij lage signaal-ruisverhoudingen ( $<5$ ) superieure prestaties vergeleken met de eerder genoemde twee-staps methoden. Bovendien houdt het algoritme op een natuurlijke manier rekening met blekingseffecten, welke vaak voorkomen in fluorescentie microscopie. Experimenten met synthetische data bevestigen dat het voorgestelde algoritme betrouwbare resultaten levert, zelfs wanneer de signaal-ruisverhouding in de data kleiner is dan 2, dit in tegenstelling tot twee andere populaire methoden waarmee wordt vergeleken. Ook wordt aangetoond dat het algoritme potentieel nauwkeuriger is dan handmatige data-analyse door ervaren menselijke waarnemers. Toegepast op echte

fluorescentie-microscopiebeeldseries van microtubuli, levert het algoritme prestaties die vergelijkbaar zijn met die van menselijke waarnemers. Dit is te verklaren doordat de laatstgenoemde experimenten beperkt waren tot het vergelijken van verdelingen en gemiddelden van bewegingsparameters, waarbij kleine lokale verschillen onopgemerkt kunnen blijven, vooral wanneer de snelheden van de objecten variëren. Instantane snelheden zijn ook per object geanalyseerd, maar konden niet kwantitatief worden gevalideerd vanwege het gebrek aan objectieve referentiedata.

Over het algemeen wordt er bij de ontwikkeling van Bayesiaanse algoritmen voor het volgen van objecten uitgegaan van slechts één type dynamisch gedrag. Hierdoor kunnen deze algoritmen falen wanneer ze worden gebruikt voor biologische toepassingen waarin complexere bewegingspatronen moeten worden geanalyseerd. Daarom wordt in Hoofdstuk 4 het in Hoofdstuk 3 ontwikkelde algoritme uitgebreid zodat het verschillende typen subcellulaire objecten met verschillende soorten bewegingspatronen kan volgen. De nauwkeurigheid van het algoritme is verbeterd door marginalisatie van de kansdichtheden en één van de toestandsvariabelen, waarvoor de optimale, analytische oplossing (het Kalman filter) wordt gebruikt. Verder is de robuustheid verbeterd door gebruik te maken van een zogenaamd jump-Markov-systeem, wat het gebruik van meerdere dynamiekmodellen voor de voorspelling van objectbeweging toelaat. Het voorgestelde algoritme is op zowel synthetische data als op echte fluorescentie-microscopiebeeldseries getest. De laatstgenoemde data komen voort uit studies naar het dynamisch gedrag van drie verschillende typen intracellulaire objecten: microtubuli, vesicles, en androgeenreceptoren. De resultaten van de experimenten met synthetische data tonen duidelijk aan dat het voorgestelde algoritme superieur presteert ten opzichte van handmatig verkregen resultaten en eerdere Bayesiaanse benaderingen, waarvan reeds werd aangetoond dat deze beter presteren dan alternatieve niet-Bayesiaanse algoritmen. De experimenten met echte data bevestigen de geldigheid van de resultaten geproduceerd door het voorgestelde algoritme. Gebaseerd op deze resultaten wordt nu in de praktijk onderzocht of de algoritmen kunnen helpen bij het oplossen van specifieke biologische vraagstukken.

Tenslotte wordt in Hoofdstuk 5 de toepasbaarheid van PF methoden op een andere biologische applicatie gedemonstreerd: de analyse van het dynamische gedrag van microtubuli *in vitro*, afgebeeld door middel van differentiële interferentiecontrast microscopie. Er wordt een nieuw algoritme voorgesteld dat “variable-rate” PF (VRPF) combineert met multischaal-trendanalyse voor het analyseren van de beweging van groeiende of krimpende uiteinden van microtubuli in 2D spatiotemporele beelden (kymografen). Het voorgestelde VRPF combineert beeldinformatie met bestaande kennis van de onderliggende dynamiek van microtubuli en is in staat het uiteinde van het microtubulus te volgen zelfs in situaties waarin snelle bewegingsveranderingen optreden. Zoals aangetoond door experimenten op synthetische data is de methode in staat de belangrijke kinematische parameters nauwkeurig te schatten. In deze experimenten is de fout in het localiseren van het uiteinde van microtubuli in kymografen gemeten, en is de invloed van deze fout op de schatting van belangrijke kinematische parameters (zoals groei- en krimpsnelheden en de frequenties van schakelen tussen groei en krimp en vice versa) bestudeerd. Vanuit de theorie is bekend dat zelfs relatief kleine fouten in de localisatie van de uiteinden van de microtubuli kunnen leiden tot grote fouten in de parameterschattingen, door het niet-lineaire verband tussen de geschatte

hellingen en de berekende snelheden. De experimentele resultaten laten inderdaad een verhoogde onzekerheid zien in de snelheidsschatting bij hogere snelheden. Toegepast op echte data produceert de voorgestelde methode parameterschattingen die overeenkomen met schattingen die handmatig zijn verkregen door ervaren biologen.

---

---

# PhD Portfolio

---

## Research Skills:

- M.Sc. degree in Electrical Engineering, Ivan Franko National University of Lviv, Ukraine, 1999
- Professional Doctorate in Engineering (PDEng) degree, Technical University of Eindhoven, the Netherlands, 2005

## In-Depth Courses:

- Knowledge driven Image Segmentation, ASCI, 2005
- Measuring Features, ASCI, 2006
- Front-End Vision and Multiscale Image Analysis, ASCI, 2007
- BioInformatics, ASCI, 2007

## International Conference Presentations:

- IEEE International Symposium on Biomedical Imaging: From Nano to Macro — ISBI 2006, Arlington, VA, USA, April 6–9, 2006
- IEEE Nonlinear Statistical Signal Processing Workshop: Classical, Unscented and Particle Filtering Methods — NSSPW 2006, Cambridge, UK, September 13-15, 2006
- IEEE International Symposium on Biomedical Imaging: From Nano to Macro — ISBI 2007, Arlington, VA, USA, April, 12–15, 2007
- Information Processing in Medical Imaging — IPMI 2007, Kerkrade, the Netherlands, July 2-6, 2007
- IEEE International Symposium on Biomedical Imaging: From Nano to Macro — ISBI 2008, Paris, France, May 14–17, 2008

**Invited Lectures and Seminars:**

- Promovendidagen 2006, Maastricht, January 26-27, 2006
- Medical Imaging Symposium for PhD-Students, University Medical Center Utrecht, January 11, 2007

**Travel Grants:**

- Student travel grant, IEEE International Symposium on Biomedical Imaging: From Nano to Macro — ISBI 2007, Arlington, VA, USA, April, 12–15, 2007

**Other:**

- Referee activities for various international scientific journals (IEEE Transactions on Medical Imaging, IEEE Transactions on Image Processing, IEEE Transactions on Biomedical Engineering, Image and Vision Computing, Sensors) and international conferences (IEEE International Conference on Image Processing (ICIP) and International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI))

---

---

# Publications

---

## Publications in International Journals:

- E. Meijering, **I. Smal**, G. Danuser, “Tracking in Molecular Bioimaging”, *IEEE Signal Processing Magazine*, vol. 23, no. 3, pp. 46–53, May 2006
- **I. Smal**, K. Draegestein, N. Galjart, W. Niessen, E. Meijering, “Particle Filtering for Multiple Object Tracking in Dynamic Fluorescence Microscopy Images: Application to Microtubule Growth Analysis”, *IEEE Transactions on Medical Imaging*, vol. 27, no. 6, pp. 789–804, June 2008
- **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, December 2008
- **I. Smal**, M. Loog, W. J. Niessen, E. Meijering, “Quantitative Comparison of Spot Detection Methods in Fluorescence Microscopy”, *submitted*
- **I. Smal**, I. Grigoriev, A. Akhmanova, W. J. Niessen, E. Meijering, “Microtubule Dynamics Analysis using Kymographs and Variable-Rate Particle Filters”, *submitted*

## Book Chapters:

- E. Meijering, **I. Smal**, O. Dzyubachyk, J.-C. Olivo-Marin, “Time-Lapse Imaging” in *Microscope Image Processing*, Q. Wu, F. A. Merchant, K. R. Castleman (eds.), Elsevier Academic Press, Burlington, MA, Chapter 15, pp. 401–440, 2008

## Publications in International Conference Proceedings:

- **I. Smal**, W. Niessen, E. Meijering, “Bayesian Tracking for Fluorescence Microscopic Imaging”, in *IEEE International Symposium on Biomedical Imaging: From Nano to Macro — ISBI 2006* (3rd international conference, held in Arlington, VA, USA, April 6–9, 2006), J. Kovačević and E. Meijering (eds.), IEEE, Piscataway, NJ, pp. 550–553, 2006



- **I. Smal**, W. Niessen, E. Meijering, “Particle Filtering for Multiple Object Tracking in Molecular Cell Biology”, in *IEEE Nonlinear Statistical Signal Processing Workshop: Classical, Unscented and Particle Filtering Methods — NSSPW 2006* (conference, held in Cambridge, UK, September 13-15, 2006), IEEE, Piscataway, NJ, pp. 1–4, 2006
- **I. Smal**, W. Niessen, E. Meijering, “Advanced Particle Filtering for Multiple Object Tracking in Dynamic Fluorescence Microscopy Images”, in *IEEE International Symposium on Biomedical Imaging: From Nano to Macro — ISBI 2007* (4th international conference, held in Arlington, VA, USA, April 12–15, 2007), J. Fessler and T. Denney (eds.), IEEE, Piscataway, NJ, pp. 1048–1051, 2007
- **I. Smal**, K. Draegestein, N. Galjart, W. Niessen, E. Meijering, “Rao-Blackwellized Marginal Particle Filtering for Multiple Object Tracking in Molecular Bioimaging”, in *Information Processing in Medical Imaging — IPMI 2007* (20th international conference, held in Kerkrade, the Netherlands, July 2-6, 2007), N. Karssemeijer and B. Lelieveldt (eds.), vol. 4584 of *Lecture Notes in Computer Science*, Springer Berlin/Heidelberg, pp. 110–121, 2007
- M. Schaap, **I. Smal**, C. T. Metz, T. van Walsum and W. J. Niessen, “Bayesian tracking of elongated structures in 3D images”, in *Information Processing in Medical Imaging — IPMI 2007* (20th international conference, held in Kerkrade, the Netherlands, July 2-6, 2007), N. Karssemeijer and B. Lelieveldt (eds.), vol. 4584 of *Lecture Notes in Computer Science*, Springer Berlin/Heidelberg, pp. 74–85, 2007
- M. Schaap, R. Manniesing, **I. Smal**, T. van Walsum, A. van der Lugt and W. J. Niessen, “Bayesian Tracking of Tubular Structures and Its Application to Carotid Arteries in CTA”, in *Medical Image Computing and Computer Assisted Intervention — MICCAI 2007* (10th international conference, held in Brisbane, Australia, October 29 – November 2, 2007), vol. 4791 of *Lecture Notes in Computer Science*, Springer Berlin/Heidelberg, pp. 562–570, 2007
- **I. Smal**, W. Niessen, E. Meijering, “A New Detection Scheme for Multiple Object Tracking in Fluorescence Microscopy by Joint Probabilistic Data Association Filtering”, in *IEEE International Symposium on Biomedical Imaging: From Nano to Macro — ISBI 2008* (5th international conference, held in Paris, France, May 14–17, 2008), J.-C. Olivo-Marin, I. Bloch and A. Laine (eds.), IEEE, Piscataway, NJ, pp. 264–267, 2008
- **I. Smal**, M. Loog, W. Niessen, E. Meijering, “Quantitative Comparison of Spot Detection Methods in Live-Cell Fluorescence Microscopy Imaging”, in *IEEE International Symposium on Biomedical Imaging: From Nano to Macro — ISBI 2009* (6th international conference, to be held in Boston, MA, USA, June 28–July 1, 2009)

---

---

# Curriculum Vitae

---

Ihor Smal was born in Lviv, Ukraine, on April 7, 1977. He received a M.Sc. degree (cum laude) in Electrical Engineering from Ivan Franko National University of Lviv, Ukraine, in 1999. From 1999 to 2002, he was a Research Scientist at the Electrical Engineering department of the same university. During that period he carried out research in the field of nonlinear and chaotic dynamical systems.

From 2003 to 2005, he was a Research Assistant (postmaster program “Mathematics for Industry”) at the department of Mathematics and Computer Science of Technical University of Eindhoven, the Netherlands. In 2005 he graduated on the project “Design and implementation of a six camera scanning unit” and was awarded a Professional Doctorate in Engineering degree (PDEng).

From Feb. 2005 to Feb. 2009 he was a Ph.D. student at the Departments of Medical Informatics and Radiology of the Erasmus University Rotterdam, the Netherlands. His research topic was tracking and motion analysis in cellular and molecular bioimaging. The project was carried out in collaboration with the Department of Cell Biology and Department of Pathology at Erasmus MC Rotterdam. The results are described in this thesis.



