SUBCELLULAR STRUCTURE MODELING AND TRACKING FOR CELL DYNAMICS STUDY

by

QUAN WEN

Presented to the Faculty of the Graduate School of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

DOCTOR OF PHILOSOPHY

THE UNIVERSITY OF TEXAS AT ARLINGTON

August 2008

To my parents

ACKNOWLEDGEMENTS

I would like to thank my supervising professor Dr. Jean Gao for constantly motivating and encouraging me, and also for her invaluable advice and help during the course of my doctoral studies. I wish to thank my academic advisors Dr. Gautam Das, Dr. Gutemberg Guerra-Filho, Dr. Manfred Huber, and Dr. Farhad Kamangar for their great helps to serve in my dissertation committee.

I also wish to thank Dr. Kate Luby-Phelps of the Department of Cell Biology at The University of Texas Southwestern Medical Center for providing all the original image data. My appreciation is also extended to all the teachers and staffs of the Department of Computer Science and Engineering who help me always.

Finally, I am very grateful to my parents, my brother and sister-in-law for their support, encouragement, and patience during the years. Special thanks also go to Darin Brezeale, Yongfang Liang, Ninad Thakoor, Xiaohui Wei, Dangxin Wu, Jiaxing Xue, and Guohua Zhang.

August 6, 2008

ABSTRACT

SUBCELLULAR STRUCTURE MODELING AND TRACKING FOR CELL DYNAMICS STUDY

QUAN WEN, Ph.D.

The University of Texas at Arlington, 2008

Supervising Professor: Jean Gao

The introduction of sensitive and fast electronic imaging devices and the development of biological methods to tag proteins of interest by green fluorescent proteins (GFP) have made a full understanding of live cell dynamics achievable. With the latest hardware technology, such as high speed laser scanning confocal microscopy (LSCM), it has now become critical to develop automatic quantitative data analysis tools to keep pace with and to fully exploit the functionalities of state-of-the-art hardware. One task of such tools is the motility analysis of subcellular structures. This dissertation provides a series of computational approaches for studying subcellular structure motility.

Firstly, a semi-automatic single object tracking approach using sequential Monte Carlo (SMC) method is developed. To achieve reliable tracking, a flow of criterion for object feature selection, matching, and evaluation criteria are designed: a grid-based minimum variance (GMV) feature selection rule, a mean minimum to maximum ratio (MMMR) similarity measurement, and the feature evaluation tests by feature convergence ratio ($FCVR$) and feature consistence ratio ($FCSR$).

Secondly, to handle complex scenario of multiple interacting subcellular structure motion, we apply reversible jump Markov chain Monte Carlo (RJMCMC) method to sample the distribution of the dimension varying joint state which is the combination of the states of multiple individual subcellular structures. Five RJMCMC moves are constructed, including object appear move, disappear move, update move, height swap move, and identity swap move. The evolution of each individual state in the joint state is sampled by the update move. In order to deal with the random appearance locations of subcellular structures, a marker residual image guided appearance model is proposed to detect the newly appearing object, and the appear move and the disappear move are applied to generate samples resulting from the new object appearance. To prevent the RJMCMC sampling from being trapped at the local maxima, the identity swap move is also constructed. The proposed RJMCMC SMC tracking approach is applied to numerous time-lapse LSCM video sequence tracking in both 2D+T and 3D+T domains.

Finally, from a perspective different from the SMC framework, we model the multiple object tracking as a bipartite graph matching problem between the consecutive image frames. To save the possible high cost of graph matching, a Markov chain Monte Carlo data association (MCMCDA) method with deletion move, switch move, and addition move is developed to approximate the optimal solution.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

CHAPTER 1

INTRODUCTION

The introduction of sensitive electronic imaging devices and the development of methods to tag proteins of interest by green fluorescent protein (GFP)[1], [2], [3] have been the drivers to live-cell study. A direct example is observing the motion of GFP expressed proteins that are transported in vesicles moving from place to place along microtubule tracks. Understanding the motility of these subcellular structures like organelles, vesicles, or mRNAs is critical to understand how cells regulate the delivery of specific proteins from the site of synthesis to the site of action at subcellular level. The knowledge of regulation and how it is deranged in various diseased or malfunctioned states will eventually lead to a better understanding of such diseases as diabetes, hypercholesterolemia, and many viral infections. At present, biologists either laboriously track a few subcellular structures by hand, or use commercially available particle tracking programs whose performance is far below expectations for various demands.

The image data sets obtained using laser scanning confocal microscope (LSCM) can be classified into two categories. The first one is the two spatial dimension time series (2D+T) image sequence, and the second one is three spatial dimension time series (3D+T) image sequence. In a 2D+T image sequence, an image at a time point is taken for the whole specimen at a certain focus plane. For a 3D+T video sequence, images at a certain time point are recorded at different heights (focus planes), which overcomes the out-of-focus limitation for 2D+T and provides a better 3D knowledge of object temporal change.

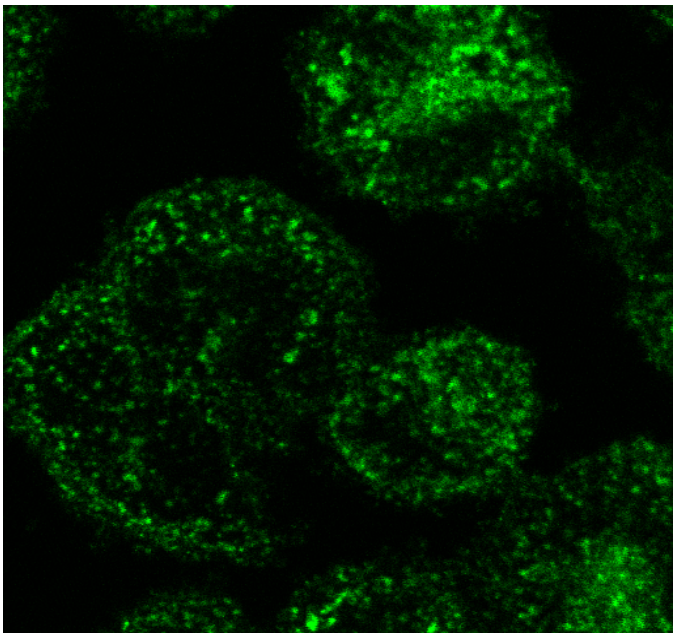


Figure 1.1. A green fluorescent protein (GFP) labeled image obtained from Laser Scanning Confocal Microscope (LSCM).

Figure 1.1 shows an example of 2D GFP image obtained from LSCM. The green regions are proteins of interest that vary in both size and intensity due to the spatial-temporal distribution of GFP in a specimen. Different from traditionally well studied video objects, subcellular structures observed by LSCM have distinctive properties:

- Variable object size, ranging from large to small;
- Changing number of objects caused by out-of-focus or dying out of fluorophore;
- Large shape deformation due to relatively long time elapse between adjacent frames;
- Lacking of stable object features resulted from inconstant fluorescence;
- Diverse motion modalities, such as Brownian motion and translational motion;
- Cluttered environment and high noise due to photo bleaching.

These characteristics pose challenges to subcellular structure tracking, to which commonly used motion analysis and object tracking techniques can not be simply applied.

Toward the goal of subcellular structure motility analysis, Chapter 2 describes a semi-automatic single object tracking approach using sequential Monte Carlo (SMC) method. Object feature selection, matching, and evaluation criteria are developed for robust tracking. This methodology benefits biologists when the motility of a specific structure is interested [4], [5], [6].

The core of the dissertation goes to Chapters 3 and 4 where the motion tracking of multiple interacting objects is dealt with. Chapter 3 describes the tracking for a non-linear, non-Gaussian, state dimension changing system in 2D+T. The reversible jump Markov chain Monte Carlo (RJMCMC) moves, including object appear move, disappear move, update move, height swap move, and identity swap move, are constructed to sample the distribution of the dimension varying joint state [7], [8], [9]. Chapter 4 presents an extension of the 2D+T object tracking to 3D+T space.

Finally, Chapter 5 [10] tackles the object tracking from a different perspective, where multiple object tracking is modeled as a bipartite graph matching problem between the consecutive image frames. To save the possible high cost of graph matching, a Markov chain Monte Carlo data association (MCMCDA) method with deletion move, switch move, and addition move, is provided to approximate the optimal solution.

CHAPTER 2

A PARTICLE FILTER APPROACH FOR SUBCELLULAR STRUCTURE TRACKING

2.1 Introduction

The introduction of high resolution electronic imaging devices and the development of methods to tag proteins of interest by green fluorescent protein (GFP) have been the drivers to live cell study. Understanding the motility of subcellular particles like organelles, vesicles, or mRNAs is critical to understand how cells regulate delivery of specific proteins from the site of synthesis to the site of action at subcellular level. The knowledge of regulation and how it is deranged in various diseased or malfunctioned states will eventually lead to a better understanding of such diseases as diabetes, hypercholesterolemia, and many viral infections. At present, biologists either laboriously track a few vesicles by hand, or use commercially available particle tracking programs whose performance needs to be amended for various demands.

2.1.1 Subcellular Structure Motility Study

There are a plethora of motility analysis methods on microscopy video data at cellular level. Compared with GFP molecules at sub-cellular level, cells have less shape deformation and displacement between frames. Therefore, cells can be tracked by the evolution of active contour methods [11], [12], [13], [14], or simply by a template matching [15]. However, motility study at sub-cellular level is more complicated. From the visual perception point of view, there are two categories of GFP molecules that are under investigation. One is small spot-like separated GFP

molecular particles, and another type is larger GFP molecular clusters (consisted of a number of neighbored spots) which are what this chapter will deal with.

Spot-like molecular particle tracking has been investigated widely in the study of molecular mobility properties [16], [17], [18], [19], [20], [21]. There are two situations in finding spot correspondence between consecutive video frames as summarized by [22]: 1) there is only one detectable object in the image frame. Therefore the tracking task is to detect the location of the object by Gaussian fitting [16], [17] or the centroid of the object [18], without the need of object detection from the image frames before or after the current one; 2) the number of detectable objects in the image is more than one, in which the image information before/after the current image frame is required for object matching [19], [20], [21]. In the work by [20] and [21], the spatial configuration of the image intensity of the protein spot is assumed to be no change. This assumption does not hold in the tracking of GFP-labelled subcellular structures. In [19], only the displacement information is used to associate the spots between the consecutive image frames to construct the trajectories.

Different from the spot-like particles, the GFP molecule clusters observed by laser scanning confocal microscopy (LSCM) have distinctive properties: 1) variable object size, ranging from large to small; 2) changing number of objects caused by transporting out of the focal plane or photobleaching of the fluorophore; 3) large shape deformation as time elapses; 4) lacking of salient features among objects in the recorded images; 5) diverse motion modalities, such as Brownian motion and non-Brownia motion, including directed motion, confined motion, and anomalous diffusion [23]; 6) cluttered and noisy image background due to detector noise. These characteristics pose challenges to subcellular structure tracking and make the previously developed object tracking techniques not directly applicable here. For an example, conventional rectangular template-based tracking [15], [24] works well in a sparse ob-

ject environment with no large shape deformation. Furthermore, large inter-frame displacement and noise may mislead the shape evolution for active contour/surface-based tracking methods [11].

2.1.2 Feature Selection, Matching, and Evaluation

To develop tracking methods that are robust in differentiating objects from the background, one of the important issues is the selection of stable and distinctive features during the tracking process. Such features may include shape corners, edges, contours, spectral parameters, color histograms, or intensity of objects. While extensive efforts have been devoted to better and robust tracking framework, limited work has been done on distinctive feature selection for tracking process. To tackle feature selection in a low-resolution and noisy image, recent endeavor has been presented by [25]. In the work, features that best distinguish object from background are defined to be the best for tracking. The method selects features by a linear combination of RGB color components and by maximizing the contrast between foreground and background. However, the unique properties of GFP particles limit the extension of this approach being applicable here. This is because GFP images are gray scale (pseudo color) and cannot provide a color space to be used for the calculation. Secondly, since the objects in GFP image are visually similar and close to each other, it is difficult to define a background separating the adjacent objects. This causes the variance ratio criterion to fail to find the highly discriminative features. Lastly, this method is not suitable to handle larger shape deformations due to the specification of a fixed rectangular box. Feature evaluation criteria have been studied by [26]. Two evaluation criteria, repeatability rate and information content, have been proposed to evaluate geometric stability and to measure the distinctiveness of features, respectively.

2.1.3 Object Tracking Techniques

Kalman filtering (KF) techniques have been most commonly-used for object tracking [24], [27], which deals with linear and uni-Gaussian modal systems. The limitations come from its incapability in handling non-linear and multi-modal systems. Methods such as extended Kalman Filter EKF, Gaussian sum filtering (GSF), unscented Kalman filter (UKF), and grids-based methods have been introduced to overcome the shortcomings of Kalman filtering. Nonetheless, none of them provides a more efficient way as the later introduced particle filter (PF) [1] [28], [29], [30], [31], [27]. Since its debut in 1993 [32], particle filter has been used as a major method to deal with nonlinear/non-Gaussian systems. For a visual tracking problem, the definition of the observation model of particle filter plays a critical role in the successful application of it [33]. Model-based observation model is one of the most common methods in visual tracking, in which the state is modeled by the geometric property of an open/closed curve or simply by a bounding box. Either the edge information along the curve [34], [35], [36], or the color histogram information within the closed-curve or bounding box [37], [38], [39] is used as observation. A combined observation of both edge and color histogram is used by [40]. When applied to GFP labeled subcellular structure tracking, the major limitations of the model-based approaches stems from lacking of edge information along the contour, not enough pixels for color histogram calculation, or unstable intensity information due to photobleaching.

2.1.4 This Work

The goal of this chapter is to design a particle filter based computational framework for subcellular structure tracking. Figure 2.1 illustrates the overall concept of

[1] Word "particle" here is a computational unit used for data sampling and is different from the ones used in subcellular "particles" that refer to subcellular structures.

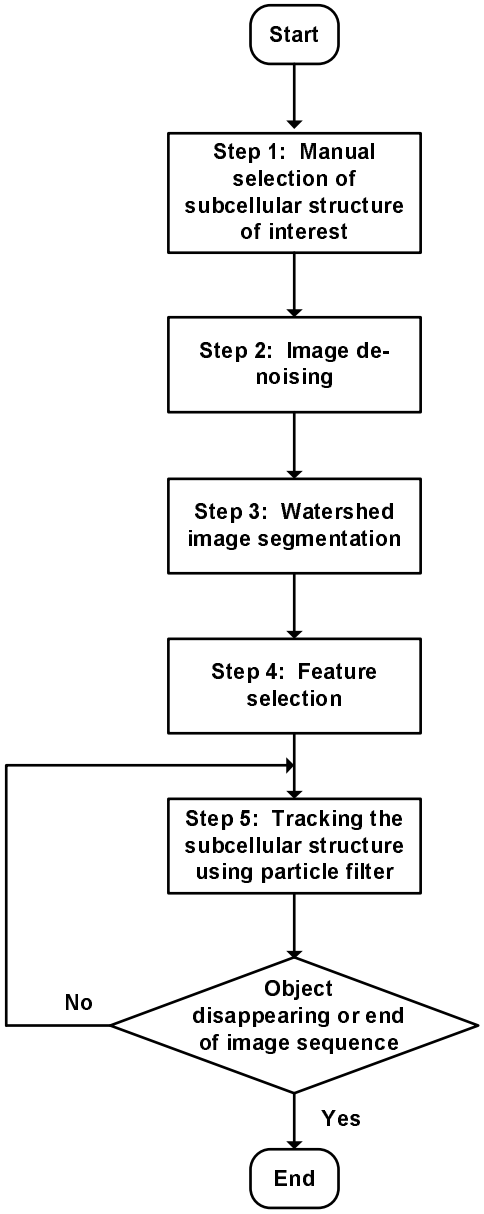


Figure 2.1.Flow chart of the particle filter tracking algorithm.

region tracking by particle filter. A novel observation model by a grid-based minimum variance (GMV) feature selection method and a mean minimum to maximum ratio (MMMR) for feature matching are first described. Then we introduce how the tracking process based on a particle filter is carried out. To evaluate the performance of

the proposed feature selection and matching techniques, two evaluation criteria, feature convergence ratio ($FCVR$) and feature consistence ratio ($FCSR$), are introduced, which conform with the Gestalt visual perception theory.

The rest of the paper is organized as follows. First, we introduce the particle filter based tracking algorithm and feature selection and matching criteria in Section 2.2. The experimental results and evaluation are presented in Section 2.3. Conclusions and future work are given in Section 2.4.

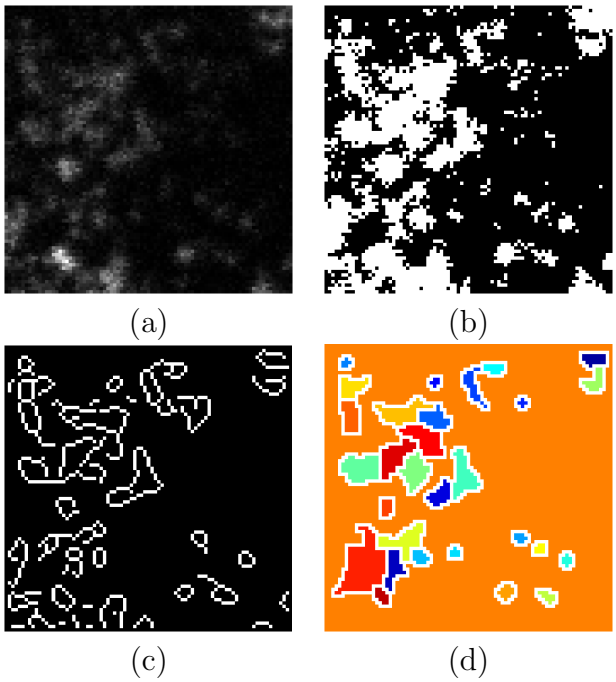


Figure 2.2. Comparison of segmentation methods. (a) Part of the original intensity image; (b) result by applying thresholding algorithm; (c) result by applying the Canny edge detection; and (d) result by Marker-guided Watershed method.

2.2 Particle Filter Based Tracking Algorithm

2.2.1 Subcellular Structure Segmentation

The tracking process starts with the specification of object of interest. A region containing the object of interest is manually selected at the beginning of the automatic tracking process. A 3×3 averaging filter $[1/9, 1/9, 1/9; 1/9, 1/9, 1/9; 1/9, 1/9, 1/9]$ is applied to the image for de-noising purpose.

As introduced in section 2.1, the GFP objects in the LSCM image sequences are unique in several aspects. Since the GFP objects in the LSCM image sequences are close to each other in a highly noisy environment without distinguishable boundaries, it is challenging to segment out the objects accurately. Furthermore, due to large shape deformation, a pre-defined fixed object shape description is not a feasible solution, too. Therefore, to segment the object of interest, proper image processing is needed. Here, we use a marker-guided watershed method to segment the object [41]. Compared with the thresholding and Canny edge detection segmentation techniques, marker-guided watershed method keeps consistence with human perception as shown in Fig. 2.2. The result of marker-guided watershed method is accurate enough for our object description and feature selection. Furthermore, the maker-guided watershed method shows highly robust performance when segmenting objects in different image frames.

2.2.2 Feature Selection by Grid-based Minimum Variance and Feature Matching by Similarity Measurement

Before we introduce the particle filter based tracking for the segmented object, we first describe the criteria used for feature selection and matching during the tracking process.

From the segmented particle of interest, intuitively all the pixels of the whole region would be utilized for the consequent object tracking. However, not every pixel conveys useful and robust information for the tracking process. The usage of the whole region is catastrophic for image sequences acquired under undesirable conditions. Therefore, the region tracking problem is often relaxed to the process of sub-region tracking stipulated by robust features. There are several methods in the literature on region feature representation, such as texture [42], histogram [43], and salient points [44]. Nonetheless, the characteristics of GFP image are unlike well studied common images. High frequency features such as edges and corners are not reliable in the GFP tracking due to the large shape deformation and noisy object environment. Furthermore, the segmentation results are not always ready for the direct application of the above methods.

Our idea is to exclude the pixels that are not suitable for robust visual tracking. Based on the Gestalt law of visual perception, we define four criteria for robust feature selection in GFP images: 1) homogeneity; 2) stability; 3) majority; and 4) proportion. For homogeneity, the features of a GFP object are defined as the pixels with the smallest intensity variance within its local support region L. The intensity variance σ_L^2 of a feature pixel with its local support is calculated as:

$$\sigma_L^2 = \frac{1}{N_L} \sum_{\vec{p} \in L} (I(\vec{p}) - \mu_L)^2, \tag{2.1}$$

where $I(.)$ is the intensity of pixel $\vec{p}$, N_L is the number of pixels in L, and μ_L is the mean intensity of local support L and calculated as: $\mu_L = 1/N_L \sum_{\vec{p} \in L} I(\vec{p})$. Considering GFP objects illuminate themselves, a stable feature point and its local support should have higher intensity than the background and other darker area of the same GFP object. This intensity stability threshold is denoted as I_h. Majority in Gestalt law means that besides the feature point itself, a larger proportion of the

local support, say larger than the majority threshold T, should belong to the GFP object. The forth criterion of proportion is from the histogram point of view, and implies that the larger region should have more feature points than the smaller one when the first three criteria are satisfied. We thus develop a grid-based minimum variance (GMV) method for feature selection by taking into consideration of both the region segmentation and reliability of each sub-region to be tracked. The method is summarized as follows:

Algorithm of GMV Feature Selection

1. Establish the stability threshold I_h, majority threshold T, and the size of each $n \times n$ sub-region, e.g. 3×3.
2. Find the minimum bounding rectangle (MBR) of the segmented object. If the length or the width of the MBR is not a multiple of n, extend it to be a multiple of n by expanding the MBR toward right or the bottom of the image with a minimum increase. The newly delineated region is called extended minimum bounding rectangle (EMBR).
3. Uniformly divide MBR or EMBR into square sub-regions with size $n \times n$.
4. Repeat for all the sub-regions of MBR or EMBR
   (a) Construct candidate features set Ω by selecting an object pixel satisfying the criteria of stability threshold I_h and majority threshold T.
   (b) If Ω is empty, go to step 4 for next sub-region.
   (c) If Ω is not empty, apply Eq. (2.1) to each pixel in Ω, and select the one with the smallest σ_L^2 as a feature point. For candidates with the same σ_L^2, the one with largest μ_L is selected. If this is still not met, the one with the largest number of object pixels in its local support is chosen.
5. End Repeat.

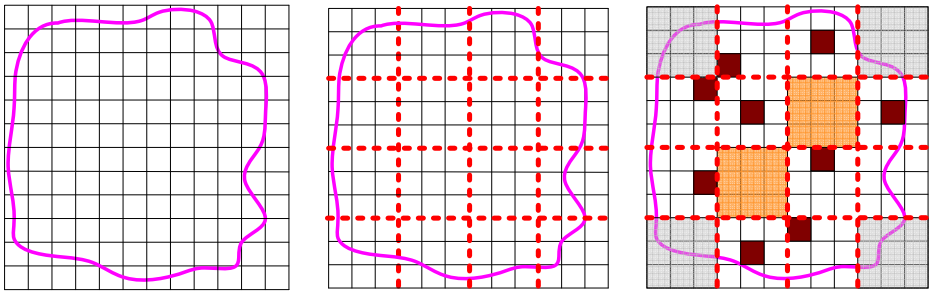


Figure 2.3. Illustration of GMV feature selection method. (a) Segmented object with its boundary (purple contour). (b) The result of applying grid lines (red doted lines) to (a). (c) Feature selection result by GMV method. Burgundy pixels are the selected feature points.

Supposing a GFP object is segmented as shown in Fig. 2.3(a), the object region is uniformly divided into sub-regions with size of $n \times n$, say 3×3, represented as red dotted squares in Fig. 2.3(b). Within each sub-region, we select all the points satisfying the Gestalt stability and majority criteria as candidates and choose the one with the lowest local support variance σ_L^2 as the feature point to represent the sub-region to comply with the homogeneity criterion. The selected feature points are shown in Fig. 2.3(c). Using the GMV feature selection, it is possible that one sub-region has no feature point extracted as shown in the two orange regions of Fig. 2.3(c). This is because the local support intensities of all the pixels in the sub-regions are less than the stability threshold I_h and violate the stability criterion. This is the desired result and is one of the advantages of GMV over simple uniformly sampling of the region for feature selection. Also, the gray-shaded sub-regions have no feature points due to the numbers of pixels of them are less than the majority threshold $T = 5$. The superiority of GMV lies in its consistence and convergency of correspondences for GFP cluster tracking, which is to be quantitatively proved in the experimental section.

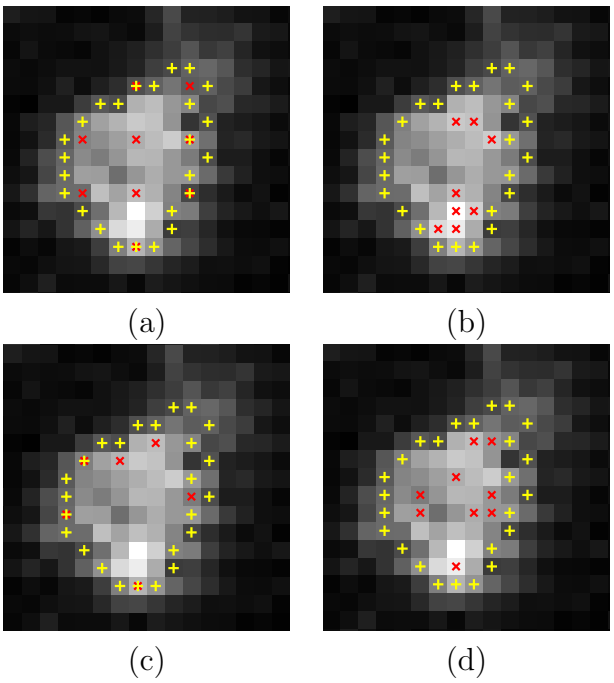


Figure 2.4. Comparison of feature selection methods. The red cross and yellow plus signs indicate the positions of the selected feature points and the boundary pixels, respectively. (a) is for USS. (b) is for MIP. (c) is for Harris corner. (d) is for GMV.

The GMV feature selection method was compared with three other often used feature selection methods, namely uniform sub-sampling (USS), maximum intensity pixel (MIP), and Harris corner detector [45]. USS is a method of uniformly sub-sampling the region of interest. MIP method uses pixels with highest intensity as features. With the number of feature pixels in MIP set to be the same as USS method, as seen in Fig. 2.4, the selected feature points of different methods have different patterns. In Fig. 2.4(a), the USS method does not consider the local pixel information such as intensity or gradient, so the selected feature pixels have no obvious patterns except they are uniformly distributed. For the MIP method, the feature points aggregate in the high intensity parts, as shown in Fig. 2.4(b). The corner pixels detected by Harris method are shown in Fig. 2.4(c). Here, the parameter k for Harris corner detection in $R = Det - kT_r^2$, the last equation of [45], is set to

be $k = 0.04$. We set the threshold value of R to be 0.01 of the maximum observed point strength. Reader please refer to [45] for the detailed descriptions of R, Det, k, and T_r^2. On the other hand, feature points of GMV tend to represent the intensity composition of the whole region.

One important aspect in object tracking is how to measure the similarity between a selected feature point and its correspondence. The mostly used similarity measurement between feature points is normalized cross-correlation (NCC). This rule works well for feature points executing translational motion and with less intensity distortion, but it is sensitive to feature points undergoing affine motion. To overcome this limitation, a different similarity measurement between feature point $\vec{p}_s$ in frame k and candidate point $\vec{p}_t$ in frame n is defined as:

$$S(\vec{p}_s, \vec{p}_t) = \frac{1}{|L|} \sum_{\vec{p} \in L} \frac{\min[I_k(\vec{p}), I_n(\vec{p} + \vec{p}_t - \vec{p}_s)]}{\max[I_k(\vec{p}), I_n(\vec{p} + \vec{p}_t - \vec{p}_s)]}, \tag{2.2}$$

where L is the local support region for $\vec{p}_s$ at time k, $|.|$ is the cardinality operation, I_k, I_n are the image functions at time k and n, respectively. The ratio of min/max measures the similarity of two pixels in the two support regions corresponding to points $\vec{p}_s$ and $\vec{p}_t$, respectively. We call $S(\vec{p}_s, \vec{p}_t)$ mean minimum to maximum ratio (MMMR). The ratio returns a value in the range of $[0, 1]$. It reaches its maximum value of one when two pixels have the same intensity, and minimum value of zero when one of the pixels is zero intensity. In the case when $\max(x, y) = 0$, we set $\min(x, y)/\max(x, y) = 1$. This ratio also gives larger similarity value to two pixels with higher intensity values comparing to lower ones with the same intensity difference. For example, if $I(\vec{p}_s) = 80$, $I(\vec{p}_t) = 100$, $I(\vec{p}_t) - I(\vec{p}_s) = 20$, the min/max ratio is 0.8. However for another pair of pixels, $I(\vec{p}_s) = 30$, $I(\vec{p}_t) = 50$, $I(\vec{p}_t) - I(\vec{p}_s) = 20$, the min/max ratio is 0.6 which is lower than the previous one. A candidate point

with the highest MMMR in the search window will be picked up as the matching feature for feature point $\vec{p}_s$.

Based on the similarity measurement, we define the distance function between points $\vec{p}_s$ and $\vec{p}_t$ as:

$$D(\vec{p}_s, \vec{p}_t) = 1 - S(\vec{p}_s, \vec{p}_t). \tag{2.3}$$

The distance function defined by Eq. (2.3) is more robust to deal with affine motions than NCC measurement, which can be seen in the experiment section.

2.2.3 Subcellular Structure Tracking by Particle Filter

For the initially segmented subcellular structure as described above, the tracking process is described by a state space model. The state of the subcellular structure at different time t can be modeled as a first order Markovian process with state $\mathbf{X}_t$, $t = 0, 1, \cdots$, and the output of the system is denoted as observation $\mathbf{Z}_t$, $t = 1, 2, \cdots$, measured from the image sequence. The dependence relationship between the states and observations is illustrated in Fig. 2.5.

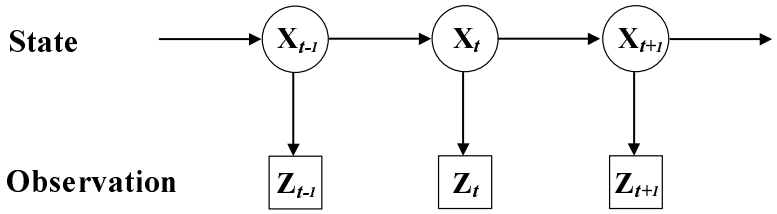


Figure 2.5.Diagram of discrete state space model.

For a Markovian process, the current state $\mathbf{X}_t$ of the subcellular structure is only conditionally dependant on its previous state $\mathbf{X}_{t-1}$, and the image observation $\mathbf{Z}_t$

is only conditionally dependant on the current state $\mathbf{X}_t$. This relationship is described by conditional probability density functions:

$$p(\mathbf{X}_0), \quad t = 0 \quad \text{Initial state} \tag{2.4}$$

$$p(\mathbf{X}_t|\mathbf{X}_{t-1}), \quad t \geq 1 \quad \text{Transition density} \tag{2.5}$$

$$p(\mathbf{Z}_t|\mathbf{X}_t), \quad t \geq 1 \quad \text{Observation density.} \tag{2.6}$$

Particle filter (PF) is used to approximate the above conditional probability density functions. It is a sub-optimal filter, also known as sequential Monte Carlo (SMC) method [27]. The main idea of PF is to recursively generate particles (samples) that approximate the probability distributions under investigation. The particles allow further estimates of the probability distributions, including minimum mean square error (MMSE) and maximum a posteriori (MAP) estimates. For instance, if we let $\mathbf{X}_{0:t} = \{\mathbf{X}_0, \cdots, \mathbf{X}_t\}$ and $\mathbf{Z}_{1:t} = \{\mathbf{Z}_1, \cdots, \mathbf{Z}_t\}$, which represent the states and the observations up to time t respectively, the posterior probability distribution of interest is $p(\mathbf{X}_t|\mathbf{Z}_{1:t})$. This can be approximated by particles as:

$$p(\mathbf{X}_t|\mathbf{Z}_{1:t}) \approx \sum_{i=1}^{N_s} w_t^i \delta(\mathbf{X}_t - \mathbf{X}_t^i), \tag{2.7}$$

where δ is the Dirac delta function, $\mathbf{X}_t^i$ is the particle with w_t^i as its normalized sample weight, and N_s is the number of samples. When the prior density function is chosen as the importance function, the sample weights of the particles can be recursively updated as:

$$w_t^i = w_{t-1}^i p(\mathbf{Z}_t|\mathbf{X}_t^i). \tag{2.8}$$

To estimate $p(\mathbf{Z}_t|\mathbf{X}_t^i)$ in Eq. 2.8, we need to consider individual segmented regions each particle $\{\mathbf{X}_t^i\}$ is associated with. This is because after the state transition process from $\{\mathbf{X}_{t-1}^i\}$ to $\{\mathbf{X}_t^i\}$, the N_s particles originally associate in the same reference region R_r at the beginning of the tracking may now be affiliated with different

regions. We denote the region a particle associated with as R^i. Without loss of generality, the observation density for a given particle can be approximated by the matched feature points distributions with each being a Gaussian distribution. Hence, $p(\mathbf{Z}_t|\mathbf{X}_t^i)$ can be estimated as:

$$p(\mathbf{Z}_t|\mathbf{X}_t^i) \propto \exp\{\sum_{\vec{p}_s \in R_r, C(g_s) \in R^i} -\lambda D^2[\vec{p}_s, C(\vec{p}_s)]\}, \tag{2.9}$$

where $C(\vec{p}_s)$ is the matching point of $\vec{p}_s$ in the region R^i, and λ is a probability normalization parameter.

The whole particle filter algorithm for tracking subcellular structure is summarized below:

Framework for Region Tracking by Particle Filter

1. Select a rectangle window and segment the subcellular structure of interest at the beginning of the tracking process. This is carried out by the marker-guided watershed method with human in the loop. Feature points inside the segmented region, R_r, are extracted by GMV algorithm. Use the mean of the coordinates of the feature points as the initial state $\mathbf{X}_0$, sample it with N_s particles and represent $\mathbf{X}_0$ as $\{\mathbf{X}_0^i\}$, $i = 1, \cdots, N_s$, each with particle weight $1/N_s$.
2. Set $\{\mathbf{X}_{t-1}^i\} = \{\mathbf{X}_0^i\}$.
3. Apply state transformation Eq. (2.5) to each particle in $\{\mathbf{X}_{t-1}^i\}$, and obtain $\{\mathbf{X}_t^i\}$.
4. Apply marker-guided watershed method to segment image frame at t (to save computation cost, only regions having at least one particle in $\{\mathbf{X}_t^i\}$ are segmented). Associate each particle with the region containing it. Thus we get R^i for each particle in $\{\mathbf{X}_t^i\}$.
5. Apply Eq. (2.9) to each pair of R^i and $\{\mathbf{X}_t^i\}$, and select the region R^i with maximum $p(\mathbf{Z}_t|\mathbf{X}_t^i)$ as the correspondent region at time t for R_r.

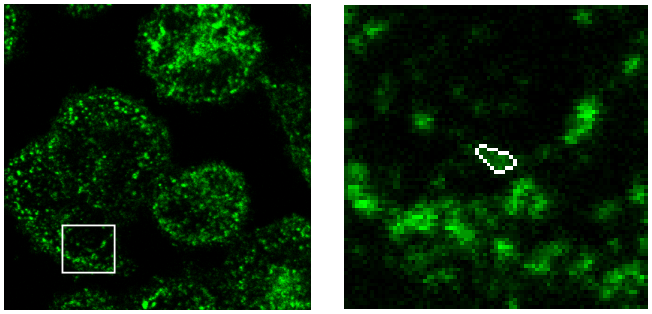
6. Update the particle weights by Eq. (2.8), and re-sample the state $\mathbf{X}_t$, we get new $\{\mathbf{X}_t^i\}$ using systematic resampling [27].
7. Set $t = t + 1$.
8. Goto step 3.

2.3 Experimental Results

In this section, we provide the experimental results from the proposed region tracking by particle filter. Our goal is to study the motility of caveolin-1-GFP which is a GFP tagged caveolin-1 protein. The motility is studied by investigating the controlling functions of microtubules and actin cytoskeleton on caveolin-1-GFP structures. CHO K1 cells were transfected with the caveolin-1-GFP. The image sequences were acquired by a Leica TCS-SP laser scanning confocal microscope with a 100X, 1.4 NA, plan-apochromatic lens using a computer controlled 488 nm argon laser to excite GFP. The images are of size 512×512 with a time interval of 1.6 second between two frames. We will first present the results of the tracking performance on two types of GFP object movement, namely non-Brownian motion and Brownian motion. The evaluation of feature selection and matching is given at last.

2.3.1 Tracking Results

In the experiment, the state is defined as $\mathbf{X}_t = [u_t, v_t, u_{t-1}, v_{t-1}]^T$, where u_t, v_t are horizontal and vertical coordinates at frame t. The state prediction by Eq. (2.5) is simplified as: $\mathbf{X}_t = \mathbf{F}\mathbf{X}_{t-1} + \mathbf{Q}_t$, where $\mathbf{F}$ = [2, 0, -1, 0; 0, 2, 0, -1; 1, 0, 0, 0; 0, 1, 0, 0], and the covariance matrix of Q_t as $\Sigma_{\mathbf{Q}}$ = [15, 0, 0, 0; 0, 15, 0, 0; 0, 0, 0, 0; 0, 0, 0, 0]. λ in Eq. (2.9) is set to be 5. The number of particles is set to be $N_s = 100$. Here we demonstrate the performance of our method by applying it to track both Brownian and non-Brownian motions.



(a) (b)

Figure 2.6. Non-Brownian motion particle selection. (a) Frame 108 where a rectangle window containing object of interest is selected; (b) The enlarged image of the selected window with the object of interest segmented by watershed algorithm.

2.3.1.1 Tracking Non-Brownian Motion

A portion of the image frame is depicted as a white rectangle in Fig. 2.6(a). The object of interest in the selected window is segmented by the watershed algorithm as the white contour in Fig. 2.6(b). Following the algorithm of region tracking by particle filter in Section 2.2, the tracking result is shown in Fig. 2.7. Since the object of interest recently appears, its intensity is much lower than its neighbors. Nonetheless, once the object is selected, our method tracked it successfully in its whole life span, despite the large shape deformations in the intermediate frames. After frame 120, the object merged with its neighbors. As can be seen from Fig. 2.7, where the tracked object is indicated by the white boundary, the shape change is dramatic in size and boundary. The imperfect segmentation result is produced by the segmentation method. The tracking trajectory of the object is shown in Fig. 2.8, where the gravity center, $[u_m, v_m]^T$, of the tracked object is used. The center is calculated by:

$$[u_m, v_m]^T = \frac{1}{N}\sum_{i=1}^{N}[u_i, v_i]^T, \tag{2.10}$$

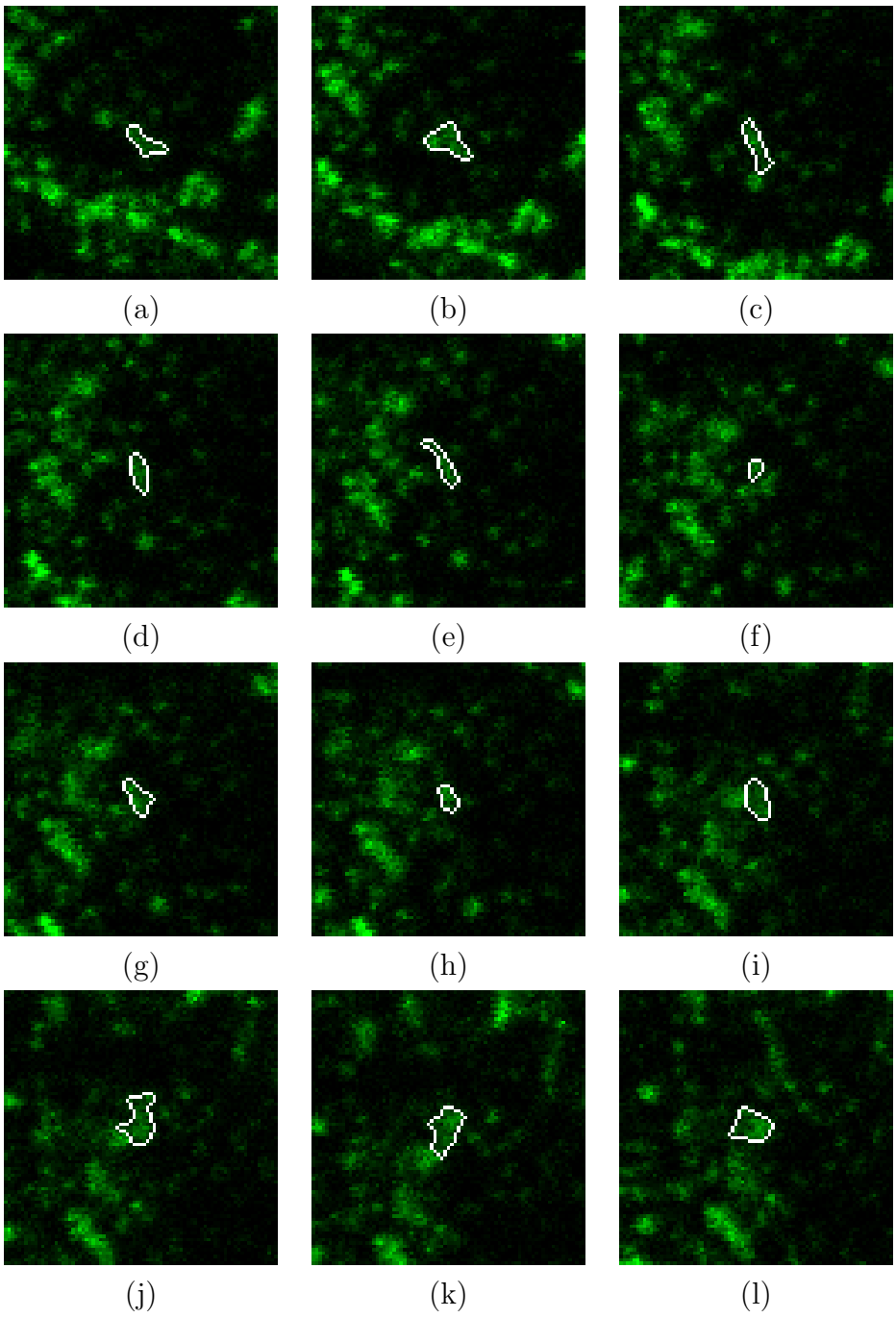


Figure 2.7. Tracking non-Brownian motion. The tracked object is marked by white contour. (a) to (l) are image frames from 109 to 120.

where $[u_i, v_i]$ is the horizontal and vertical coordinates of the ith pixel that belongs to the tracked cluster, and N is the number of the pixels in the tracked region.

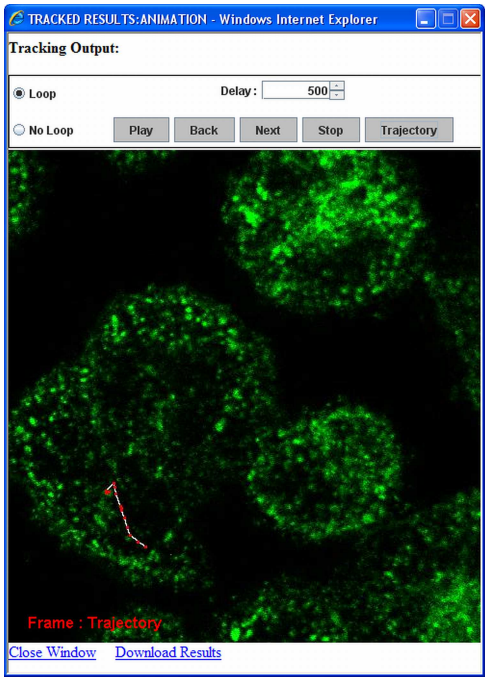
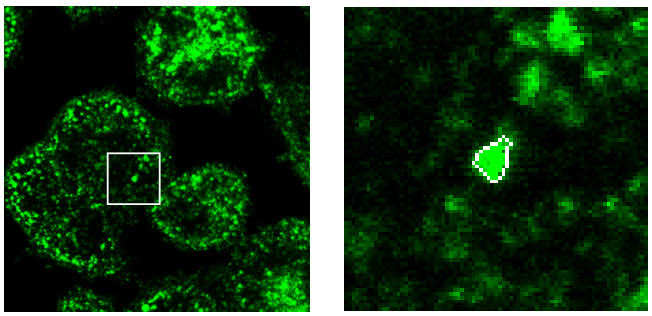


Figure 2.8.The trajectory of the non-Brownian motion particle.

2.3.1.2 Tracking Brownian Motion

The tracking performance of our method on Brownian motion is illustrated in Fig. 2.9 and Fig. 2.10. As shown in Fig. 2.9(a), the object in selected window is to be tracked. Its enlarged image is viewed in Fig. 2.9(b). Although photo-bleaching happened in the whole video sequence and the neighbor and shape of the tracked object changed a lot, our method successfully tracked the object of interest from frame 1 to the last frame 220 of the video sequence, as shown in Fig. 2.10. Same as before, the imperfect segmentation result is caused by watershed method. Figure 2.11

draws the trajectory of the object. As predicted, localized, tethered movement can be seen.



(a) (b)

Figure 2.9. Brownian motion particle selection. (a) Frame 1 where a white rectangle region delimitating the object of interest is selected; (b) The enlarged image of the white square region in (a) with the object of interest segmented as the white contour.

2.3.2 Feature Evaluation Methods

As mentioned in Section 2.1, the convergence and consistence of selected features are important in tracking subcellular structures. The discrimination and robustness of features are two major concerns. We interpret the discrimination and robustness of features points selected in region tracking as feature convergence ratio ($FCVR$) and feature consistence ratio ($FCSR$). The idea of $FCVR$ is illustrated in Fig. 2.12. Suppose there are four regions A, B, C, and D in frame $t-1$ as shown in Fig. 2.12(a), we denote them as R_A, R_B, R_C, R_D, respectively. Similarly, for frame t as shown in Fig. 2.12(b), there exist four regions represented by R^1, R^2, R^3, R^4. The region of interest is R_A with N_{fA} feature points. The correspondence of each feature point in region R_A is searched in frame t and scatters in R^1, R^2, R^3, R^4, with each region having a number of N_{f1}, N_{f2}, N_{f3}, N_{f4} feature points, respectively. Suppose

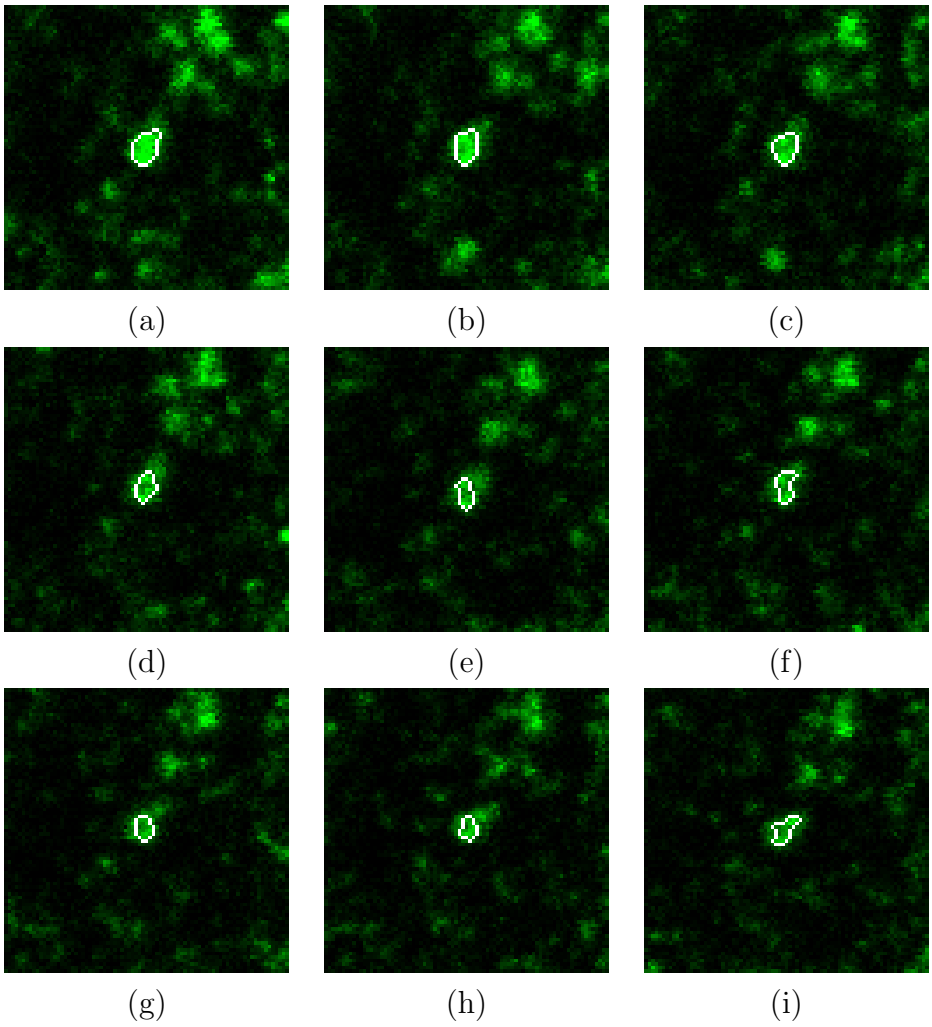


Figure 2.10. Tracking Brownian motion. The tracked object is depicted by white contour. (a) to (i) are image frames with number 30, 60, 90, 120, 150, 165, 180, 210, and 220.

region R_1 is the correspondent region with respect to region R_A determined by the maximum likelihood in particle filter, we define the $FCVR$ for region R_A as:

$$FCVR = \frac{N_{f1}}{N_{fA}}. \tag{2.11}$$

For the case in Fig. 2.12, $N_{f1} = 6, N_{fA} = 10$, and $FCVR = 0.6$. The higher the $FCVR$, the better the feature selection method.

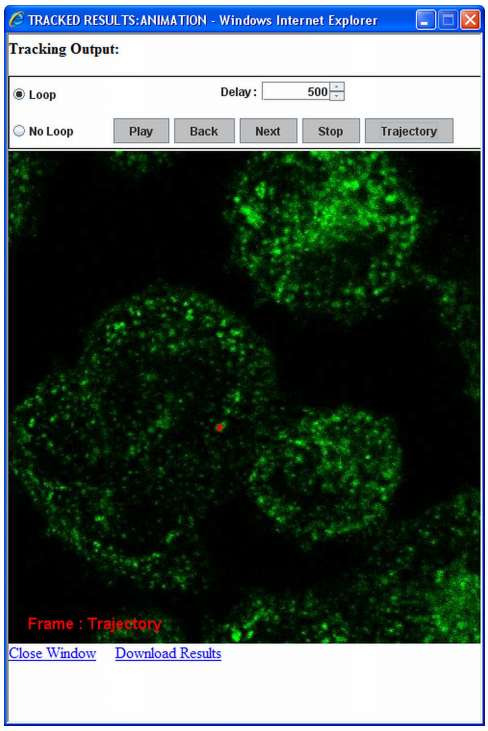
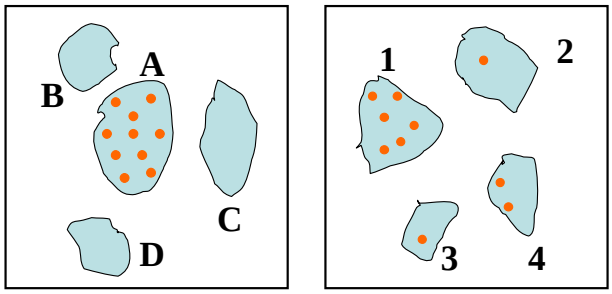
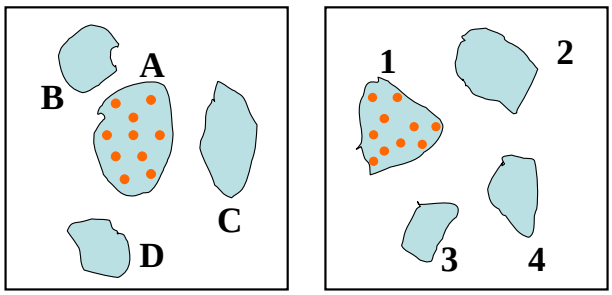


Figure 2.11.The trajectory of the Brownian motion particle.



(a) (b)

Figure 2.12. Illustration of feature convergence. (a) is the frame at time *t-1*. (b) is the frame at time *t*. Region A in (a) has ten feature pixels indicated by orange dots. The feature pixels' correspondences scatter in different regions in (b).



(a) (b)

Figure 2.13. Illustration of feature consistence. (a) is the frame at time *t-1*. (b) is the frame at time *t*. Region A in (a) has ten feature pixels indicated by orange dots. Its feature pixels' correspondences indicated by orange dots in (b) are confined in one region by restricted search.

On the other hand, *FCSR* is defined as the mean similarity value between the correspondent points in the correspondent region and their original feature points of the tracked region. Let us use Fig. 2.13 for illustration. In Fig. 2.13, suppose region R^1 is the correspondent region of R_A. The *FCSR* is defined as:

$$FCSR = \frac{1}{N_{fA}} \sum_{\vec{p}_s \in R_A, C(\vec{p}_s) \in R_1} S[\vec{p}_s, C(\vec{p}_s)] \tag{2.12}$$

where $S(.)$ is the MMMR measurement as introduced in Eq. (2.2), $C(\vec{p}_s)$ is the correspondence of feature point $\vec{p}_s$ in region R^1. Same as *FCVR*, the higher the *FCSR*, the better the feature selection method.

As can be seen from the definitions of *FCVR* and *FCSR*, we do not emphasize the discrimination and robustness of each individual feature. Instead, what we are most interested in is the properties of the whole feature set residing in the region of interest.

Figure 2.14 shows the evaluation result of the previous non-Brownian motion tracking when using proposed GMV feature selection, USS, and Harris corner detector. In the case that one method fails tracking the object correctly, we manually correct it to continue the tracking process. The means of *FCVR* and *FCSR* for GMV

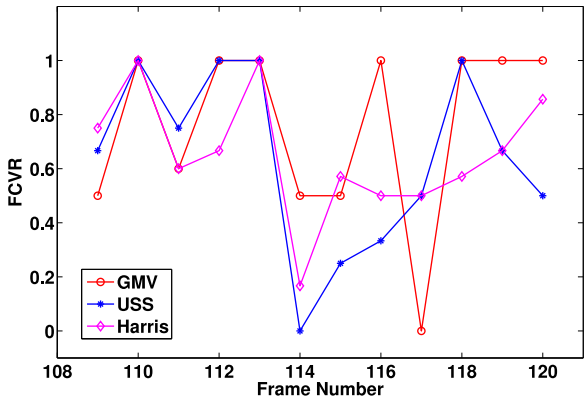
are 0.76 and 0.74, respectively, while for USS they are 0.64 and 0.70. For Harris corner detector, they are 0.65 and 0.70. This indicates the GMV method is the best for feature selection among the three.

We also validate the performance of feature selection by comparing the final tracking results. Given the human interactively segmentation region from watershed as ground truth, we validate the tracking results by checking the similarity measurements. This is reflected as the summed distance function D^2 defined in Eq. 2.3 for all feature points between two frames. We use $\sum D^2$ $difference$ between the region with the smallest $\sum D^2$ excluding the ground truth region and the ground truth region as an indication of correct tracking or not. If the $\sum D^2$ $difference$ is larger than zero, it means that the tracking is correct. Or else, the tracking is wrong, since the decision for the tracked region selection in Eq. 2.9 is made based on the smallest $\sum D^2$. In other words, whenever the $\sum D^2$ $difference$ is less than zero, it indicates a wrong decision was made.

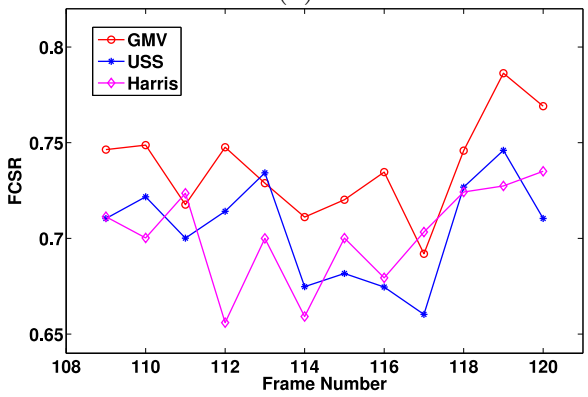
Figure 2.14(c) shows the normalized $\sum D^2$ $difference$ for three different feature selection methods. As can be seen, our method keeps all the normalized $\sum D^2$ $difference$ values above zero in the life time of the GFP labeled non-Brownian motion particle tracking, while the USS method fails the tracking in frames 114 and 117. Although Harris method gives relatively comparable or even better result in terms of overall normalized $\sum D^2$ $difference$, it fails in frame 114 indicated by less than zero normalized $\sum D^2$ $difference$.

To evaluate the similarity measurement MMMR, we compare it with the most used normalized cross-correlation (NCC). NCC is defined as:

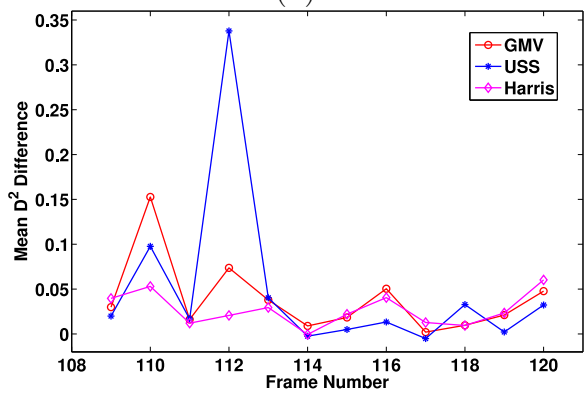
$$NCC=\frac{\sum_{\vec{p}\in L}[I_k(\vec{p})-\bar{I}_k][I_n(\vec{p}+\vec{m})-\bar{I}_n]}{\left[\sum_{p\in L}[I_k(\vec{p})-\bar{I}_k\,]^2\sum_{p\in L}[I_n(\vec{p}+\vec{m})-\bar{I}_n]^2\right]^{\frac{1}{2}}}, \tag{2.13}$$



(a)

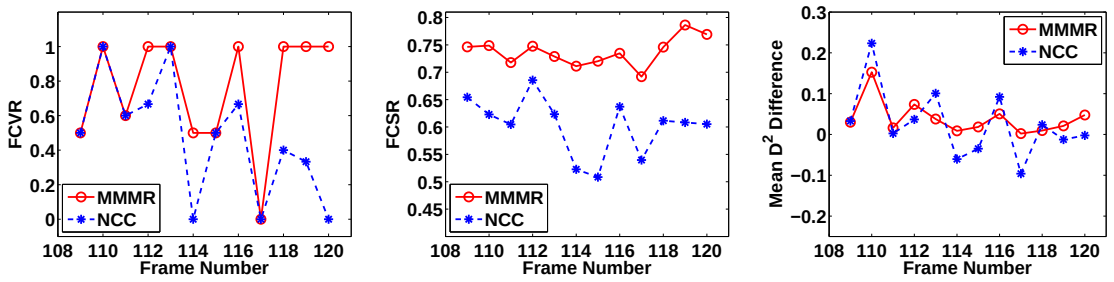


(b)



(c)

Figure 2.14. Evaluation results of the feature selection methods. (a) is for FCVR. (b) is for FCSR. (c) is for the mean D^2 difference.



(a) (b) (c)

Figure 2.15. Evaluation results of the similarity measurements. (a) is for FCVR. (b) is for FCSR. (c) is for the mean D^2 difference.

where L is the local support or template of a point of interest, $\vec{p}$ is a point in L, I_k and I_n represent image functions of frames k and n, respectively, $\vec{m}$ is the displacement vector, $\bar{I}_k$ and $\bar{I}_n$ are the mean values of the image function within the local support template. Here we use the absolute value of NCC as our similarity measurement. This means that both the 1 and -1 of original NCC will have the same similarity value. Thus the absolute value NCC is more robust to pixel doing motion with local spatial change than original NCC. In the rest of this section, we use NCC instead of absolute value of NCC, for simplicity.

Similar to GMV and USS, the comparison results between MMMR and NCC are illustrated in Fig. 2.15. Comparing to $FCVR$ of 0.76 and $FCSR$ of 0.74 by MMMR, the average of $FCVR$ and $FCSR$ by NCC feature matching in each frame time are 0.47 and 0.60, respectively. The better performance can also be seen in Fig. 2.15(c). The normalized $\sum D^2$ $difference$ values of NCC are below zero in frames 14, 15, 17, 19, and 20, which indicate the failure of the tracking process.

2.4 Conclusions

In this chapter, we presented a particle filter based framework for GFP labeled subcellular structure tracking in confocal microscopy image sequences. This frame-

work is supported by a novel GMV feature selection method and an MMMR feature matching method. The experimental results show the method can successfully track both Brownian and Non-Brownian motions in a cluttered environment and withstand a wide range of object shape changes. The robustness of the framework conveys the potential to the analysis of other scientific image sequence data such as total internal reflection fluorescence microscope (TIRFM) video sequences.

For the future work, we plan to extend the current method to automatic object detection and multi-object tracking (MOT). As the general problem of MOT, we will tackle situations like: 1) the object is split into several ones with or without dramatic intensity change; 2) the merging of several protein clusters; and 3) object moves fast in the z direction (the object intensity difference between the in-focus and the out-of-focus confocal images is large), where the region information changes spectacularly and feature points alone are not enough for the tracking problem.

CHAPTER 3

TRACKING MULTIPLE INTERACTING SUBCELLULAR STRUCTURES IN 2D+T

3.1 Introduction

The introduction of sensitive and fast electronic imaging devices and the development of biological methods to tag proteins of interest by green fluorescent proteins (GFP) have made a full understanding of live cell dynamics achievable. With the latest hardware technology, such as high speed confocal microscopy, it has now become critical to develop automatic quantitative data analysis tools to keep pace with and to fully exploit the functionalities of state-of-the-art hardware. At present, biologists either laboriously track a few particles by hand frame to frame, or use commercially available particle tracking programs whose performance is far below expectations. We lack a general computational methodology on how to represent, track, and model the motility of subcellular structures to meet the essential research needs from the biology research community.

An example of GFP labeled image obtained from laser scanning confocal microscope (LSCM) is shown in Figure 1.1. The green parts of the image are subcellular structures of interest, varying both in size and intensity caused by the different three dimension distribution of GFP inside the cell. The highly noisy background can be seen as scattered green speckles.

The technical challenges for subcellular structure mobility analysis come from small object size, nearly homogeneous object visual appearance, and objects being close to each other. Research has been carried out on cell segmentation and tracking

using simple model-free heuristic intensity thresholding [15] or model-based active contour tracking [14]. However little effort has been focused on subcellular tracking, which is much more challenging than cell tracking since cells have restricted movement and larger object size. The commonly developed wealth of object tracking techniques can not be directly applied here. For instance, conventional template-based feature matching methods work well in a sparse object environment and are restricted to non-deformable object tracking [15], [24]. The curve/surface evolution based tracking methods are susceptible to noise and inter-frame object displacement [11]. Furthermore, how to model object interaction during the tracking process is missing in current research work.

Another challenge of subcellular structure motility study is the number of objects of interest. Instead of one object moving, there may be dozens of subcellular structures moving simultaneously in the cell with a high chance of interacting with each other and object appearing/disappearing. How to associate the tracked objects with observed objects depends on the trajectory association approach. Two commonly used algorithms are multiple-hypothesis tracking (MHT) [46] and joint probabilistic data association (JPDA) [47]. Original MHT algorithm proposed by [46] decomposed the tracking problem into the state estimation and the data association components. The data association is NP-hard though attempts have been made to generate k-best hypothesis in polynomial time [48]. JPDA algorithms aim to find the state estimate by evaluating the measurement-to-track association probabilities, in which the data association can be modeled as random variables and jointly estimated by Expectation-Maximization (EM) iterations [49]. Classical MHT based methods need to handle exponential computation complexity, and JPDA methods face the problem of combinatorial complexity [50]. Furthermore, both algorithms generally do not model the interaction among objects. Certain work has targeted to model the

interaction between multiple objects during the tracking process [51], [52], [53]. The basic idea is to apply Markov random field (MRF) [54] as a penalty term to keep the objects from touching each other. By doing that, the tracker will not associate different trajectories to the same observation. As mentioned by [52], this method can not handle situations when objects really overlap with each other, and will lose track of the object.

There have been several algorithms on modeling the variable number of objects during tracking, such as finite set statistics (FISST) [55] and trans-dimensional sequential Monte Carlo (TD-SMC) [56], with the inefficiency in handling both large number of objects and large changes in the object number. The reversible jump Markov chain Monte Carlo (RJMCMC) method by [57] provides a mechanism solving the comparison between trans-dimensional probabilities, and has been applied to multiple object tracking [51], [52]. However, the assumption that object will appear/disappear in the fixed regions of the image scene is not applicable in our subcellular image sequence, where this can happen in any part of the image scene.

To overcome the deficiencies in current methods for multiple interacting object tracking, we present a tracking framework based on sequential Monte Carlo (SMC) method and approximate the joint state distribution at different times using RJMCMC sampling method. To solve the problem of handling objects overlapping, we introduce an extra dimension to augment object state from 2D plane to 3D space, and define a height swap move for the RJMCMC sampling process to traverse the possible scenarios of interaction between the objects. The restriction on the location of objects appearing or disappearing is also removed by introducing an appearance model from segmentation and treating the state of the background as part of the joint state. Experiment results on synthetic and real confocal videos show that by

modeling the object interaction in 3D space, our method can detect object splitting, merging, appearing, disappearing, and overlapping.

The rest of the chapter is organized as follows. A brief introduction to the SMC method is put in Section 3.2, followed by the joint state modeling of variable number of 2D interacting subcellular structures in Section 3.3. The RJMCMC sampling method with a height swap move is detailed in Section 3.4. The 2D subcellular structure representation and measurement are presented in Section 3.5. The 2D marker residual image guided appearance model and the 2D observation model are introduced in Section 3.6 and Section 3.7, respectively. We summarize the whole algorithm of SMC method for 2D subcellular structure tracking in Section 3.8. The experimental results and conclusions are given in Section 3.9 and Section 3.10, respectively.

3.2 Sequential Monte Carlo Methods for Bayesian Estimation

3.2.1 State Space Model

For a system described by Markovian state $\mathbf{X}_t$, $(t = 0, 1, \cdots)$, and observation $\mathbf{Z}_t$, $(t = 1, 2, \cdots)$, the dependence relationship between the states and observations can be illustrated in Figure 2.5. The current state $\mathbf{X}_t$ is conditionally dependent only on its previous state $\mathbf{X}_{t-1}$, and the observation $\mathbf{Z}_t$ is conditionally dependant on the current state $\mathbf{X}_t$. This relationship can be described as conditional probability density functions:

$$p(\mathbf{X}_0), \quad t = 0, \quad \text{Initial state} \tag{3.1}$$

$$p(\mathbf{X}_t|\mathbf{X}_{t-1}), \quad t \geq 1, \quad \text{State transition density,} \tag{3.2}$$

$$p(\mathbf{Z}_t|\mathbf{X}_t), \quad t \geq 1, \quad \text{Observation density.} \tag{3.3}$$

Based on the state space model in Figure 2.5, we define $\mathbf{X}_{0:t} \triangleq \{\mathbf{X}_0, \cdots, \mathbf{X}_t\}$ and $\mathbf{Z}_{1:t} \triangleq \{\mathbf{Z}_1, \cdots, \mathbf{Z}_t\}$, which represent the time series of states and the observations

up to current time t, respectively. Given the posterior density $p(\mathbf{X}_{t-1}|\mathbf{Z}_{1:t-1})$ at time $t-1$, the predicted density of prior density can be propagated from time $t-1$ to t via the Chapman-Kolmogoroff equation,

$$p(\mathbf{X}_t|\mathbf{Z}_{1:t-1}) = \int p(\mathbf{X}_t|\mathbf{X}_{t-1}) p(\mathbf{X}_{t-1}|\mathbf{Z}_{1:t-1}) d\mathbf{X}_{t-1}. \tag{3.4}$$

Once the new observation variable $\mathbf{Z}_t$ is available at time t, the prediction density can be updated as:

$$p(\mathbf{X}_t|\mathbf{Z}_{1:t}) \;=\; \frac{1}{K} p(\mathbf{Z}_t|\mathbf{X}_t) p(\mathbf{X}_t|\mathbf{Z}_{1:t-1}), \tag{3.5}$$

where $K = \int p(\mathbf{Z}_t|\mathbf{X}_t) p(\mathbf{X}_t|\mathbf{Z}_{1:t-1}) d\mathbf{X}_t$ is a normalizing factor.

3.2.2 Sequential Monte Carlo Methods

In the high dimensional case, the system is often non-linear and multimodal, therefore, it is intractable to determine equations (3.4) and (3.5) analytically. Sequential Monte Carlo methods are often applied to approximate the true state density using a set of random samples. The density approximated by N samples is represented as $\hat{p}(\mathbf{X}_{t-1}|\mathbf{Z}_{1:t-1})$ with:

$$\hat{p}(\mathbf{X}_{t-1}|\mathbf{Z}_{1:t-1}) = \sum_{i=1}^{N} w_{t-1}^{(i)} \delta(\mathbf{X}_{t-1} - \mathbf{X}_{t-1}^{(i)}), \tag{3.6}$$

where $w_{t-1}^{(i)}$ is the associated weight related to sample $\mathbf{X}_{t-1}^{(i)}$, and is normalized such that $\sum_{i=1}^{N} w_{t-1}^{(i)} = 1$.

Plugging equation (3.6) into equation (3.4), the approximated prediction density, $\hat{p}(\mathbf{X}_t|\mathbf{Z}_{1:t-1})$, can be written as:

$$\hat{p}(\mathbf{X}_t|\mathbf{Z}_{1:t-1}) = \sum_{i=1}^{N} p(\mathbf{X}_t|\mathbf{X}_{t-1}^{(i)}) w_{t-1}^{(i)}. \tag{3.7}$$

By further applying equation (3.7) to equation (3.5), we get $\hat{p}(\mathbf{X}_t|\mathbf{Z}_{1:t})$, the approximation of the update density:

$$\hat{p}(\mathbf{X}_t|\mathbf{Z}_{1:t}) \propto p(\mathbf{Z}_t|\mathbf{X}_t) \sum_{i=1}^{N} p(\mathbf{X}_t|\mathbf{X}_{t-1}^{(i)}) w_{t-1}^{(i)}. \tag{3.8}$$

Equation (3.7) indicates the prediction density can be represented as a mixture of densities from the N samples. The approximated posterior density from equation (3.8) will be further sampled to get new particles $\mathbf{X}_t^{(i)}, (i = 1, \ldots, N)$, with new weights. The whole procedure iteratively goes on during the Monte Carlo simulation process. Typically, there are two major methods to draw samples from $\hat{p}(\mathbf{X}_t|\mathbf{Z}_{1:t})$, namely importance sampling and Markov chain Monte Carlo (MCMC) [58]. Importance sampling is widely used in particle filtering [59] applications, and suffers from sample impoverishment and degeneracy problems [27]. For our multiple object tracking problem, to deal with the changing number of subcellular structures, a method based on reversible jump Markov chain Monte Carlo (RJMCMC) is designed to sample from equation (3.8). More details will be presented in Section 3.4 .

3.3 Modeling Variable Number of 2D Interacting Subcellular Structures

3.3.1 Joint State Space Representation

The joint state $\mathbf{X}$ of multiple subcellular structures at time t is denoted as:

$$\mathbf{X}_t = \{\mathbf{X}_{t.i} | i \in n_t\}, \tag{3.9}$$

where n_t, with its cardinality $|n_t| \geq 1$, is the set of object identity number indicating which objects contribute to represent the joint state. The ith subcellular structure of interest is represented as a rectangular bounding box with state $\mathbf{X}_{t.i} = (l, w, \theta, x, y, z)^T$ which describes the box size with length l and width w, orientation angle θ specified by the major eigenvector of object region, and bounding box center position (x, y, z). Here z is an augmented variable to model the interaction between objects.

To include the background information more naturally into the calculation of the observation likelihood, unlike work from [52] and [51], we treat the background as an object in our joint state model, and introduce its state at time t as $\mathbf{X}_{t.0} = (l_0, w_0, \theta_0, x_0, y_0, z_0)$. Since the state of the background normally does not change in the same video sequence, it can be treated as a constant vector, and the background state transition probability therefore is:

$$p(\mathbf{X}_{(t+1).0}|\mathbf{X}_{t.0}) = p(\mathbf{X}_{(t+1).0} = (l_0, w_0, \theta_0, x_0, y_0, z_0)^T) = 1. \tag{3.10}$$

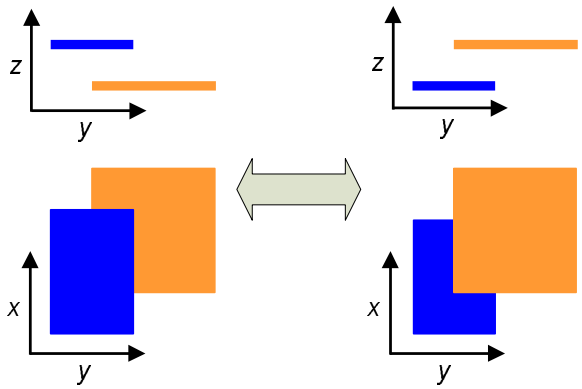


Figure 3.1. Illustration of changing the overlapping relationship between two objects by swapping their z coordinates.

3.3.2 Object Interaction Model

The idea of modeling the interaction between objects as the overlapping relationship in the z direction of the image is illustrated in Figure 3.1, where there are two overlapped objects (an orange square and a blue rectangle shown in 2D dimension). In the left part of Figure 3.1, the orange square is under the blue rectangle as shown in the left z-y plane. In the right part of Figure 3.1, the z coordinates of

the two objects are swapped, and the orange square is above the blue rectangle. The coordinate change in the z direction causes the difference of the observed images in the x-y plane. This change will be detected by a height swap move in RJMCMC that traverses the possible overlap combinations between the overlapped objects. More description will be provided in Section 3.4. Without loss of generality, we set $z_0 = 0$, which is part of the background state vector, and z of other individual object state will be always larger than z_0. The observation of the background can be easily measured by eliminating observations of other objects on top of it based on the object segmentation result.

3.3.3 Joint State Transition Density

Given state parameter vector $\mathbf{X}_t$ and object identity indicator set n_t, the joint distribution is denoted as $p(n_t, \mathbf{X}_{n_t})$. The general state transition density function $p(\mathbf{X}_{n_t}|\mathbf{X}_{n_{t-1}})$ can be re-written as $p(n_t, \mathbf{X}_{n_t}|n_{t-1}, \mathbf{X}_{n_{t-1}})$ and is further factorized as following:

$$p(n_t, \mathbf{X}_{n_t}|n_{t-1}, \mathbf{X}_{n_{t-1}}) = p(\mathbf{X}_{n_t}|n_t, n_{t-1}, \mathbf{X}_{n_{t-1}})p(n_t|n_{t-1}, \mathbf{X}_{n_{t-1}}), \tag{3.11}$$

where $p(n_t|n_{t-1}, \mathbf{X}_{n_{t-1}})$ is the probability distribution of n_t conditioned on n_{t-1} and $\mathbf{X}_{n_{t-1}}$, modeling the change of the number of the objects, $p(\mathbf{X}_{n_t}|n_t, n_{t-1}, \mathbf{X}_{n_{t-1}})$ models the joint state parameters of all the objects in set n_t. Simplifying by introducing z to each individual state $\mathbf{X}_{t.i}$, we can assume independence between the individual state of each object, and delay the object dependence check till the observation measurement stage. Thus $p(\mathbf{X}_{n_t}|n_t, n_{t-1}, \mathbf{X}_{n_{t-1}})$ can be factorized as:

$$p(\mathbf{X}_{n_t}|n_t, n_{t-1}, \mathbf{X}_{n_{t-1}}) \triangleq \prod_{j \in \mathcal{B}_t} p(\mathbf{X}_{n_t.j}) \prod_{i \in \mathcal{S}_t} p(\mathbf{X}_{n_t.i}|\mathbf{X}_{(n_{t-1}).i}), \tag{3.12}$$

where $\mathcal{S}_t = n_t \cap n_{t-1}$ is the set of objects at time $t-1$ that remain active at time t, $\mathcal{B}_t = n_t \setminus n_{t-1}$ is the set of objects that are not in set n_{t-1}. Probability distribution $p(\mathbf{X}_{n_t.j})$ will be discussed in Section 3.4.3.

3.4 RJMCMC Method for 2D+T Tracking Sample Generation

3.4.1 Acceptance Ratio For RJMCMC in SMC Framework for 2D+T Tracking

Markov chain Monte Carlo (MCMC) method is commonly used for sampling the distribution of high dimensional data. The idea is to construct a Markov chain (MC) transition kernel that has the distribution of interest as MC's invariant distribution. Metropolis-Hastings (MH) algorithm is often used to construct the MC transition kernel [60]. The general form of MH algorithm acceptance ratio is as:

$$\alpha(X, Y) = \frac{\pi(Y)q(X|Y)}{\pi(X)q(Y|X)}, \tag{3.13}$$

where X is the current sample, Y is the proposed sample, $\pi(.)$ is the density distribution of interest, and $q(.|.)$ is the proposal distribution. MH algorithm works only in the case where the dimensions of X and Y are the same. In our subcellular structure tracking problem, the number of the tracked objects changes. So does the dimension of joint state $\mathbf{X}_{n_t}$. To deal with the sampling problem of MH algorithm in changing dimensionality, Green [57] proposed the RJMCMC method as an extension of MCMC. The main idea of RJMCMC method is to match the dimension of the current sample and the proposed sample by using auxiliary variable vectors, say $\mathbf{U}$ and $\mathbf{U}'$, and fit them into the general form of MH algorithm in equation (3.13). Using the posterior distribution $p(n_t, \mathbf{X}_{n_t}|\mathbf{Z}_{1:t})$ as the target distribution and applying *Bayes'* theorem, the general form of RJMCMC acceptance ratio for SMC method is as:

$$\alpha(n'_t,\mathbf{X}'_{n_t};n_t,\mathbf{X}_{n_t}) = \frac{p(\mathbf{Z}_t|n'_t,\mathbf{X}'_{n_t})p(n'_t,\mathbf{X}'_{n_t}|\mathbf{Z}_{1:t-1})p_m q_{m'}(n_b,\mathbf{X}_{n_t};n'_t,\mathbf{X}'_{n_t})}{p(\mathbf{Z}_t|n_t,\mathbf{X}_{n_t})p(n_t,\mathbf{X}_{n_t}|\mathbf{Z}_{1:t-1})p_m q_m(n'_b\mathbf{X}'_{n_t};n_t,\mathbf{X}_{n_t})}\left|\frac{\partial(\mathbf{X}'_{n_t},\mathbf{U}')}{\partial(\mathbf{X}_{n_t},\mathbf{U})}\right|, \tag{3.14}$$

where $\mathbf{U}$ and $\mathbf{U}'$ are the auxiliary random variable vectors guarantying that the mapping from $(\mathbf{X}_{n_t}, \mathbf{U})$ to $(\mathbf{X}'_{n_t}, \mathbf{U}')$ is a one-to-one mapping, which is so called dimension matching between $\mathbf{X}_{n_t}$ and $\mathbf{X}'_{n_t}$. p_m is the move specified probability and q_m is the proposal function for $\mathbf{U}$, where m, $m \in \{d, a, u, s, h\}$, represents the move types with d, a, u, s, and h corresponding to disappear move, appear move, update move, identity swap move, and height swap move, respectively. The sum of the move probabilities is set to be one as $p_d + p_a + p_u + p_s + p_h = 1$. The last term of equation (3.14) is the Jacobian of the one-to-one mapping from $(\mathbf{X}_{n_t}, \mathbf{U})$ to $(\mathbf{X}'_{n_t}, \mathbf{U}')$. In this chaper, we present how to design both the trans-dimensional and within-dimensional moves of RJMCMC in a way such that the Jacobian term is always equal to one. Reader please see Appendix A for details.

3.4.2 RJMCMC Move Proposals for 2D+T Tracking

In our proposed RJMCMC sampling method, there are five RJMCMC moves defined, namely disappear move, appear move, update move, identity swap move, and height swap move. Disappear move and appear move are trans-dimensional moves in which the dimension of the joint state $\mathbf{X}_{n_t}$ will change. In the disappear move, we propose a new joint state by deleting the state of one subcellular structure from the current joint state. On the other hand, the appear move proposes a new joint state by adding the state of one subcellular structure to the current joint state. The rest three move modalities are within-dimensional moves. For the update move, we propose a new joint state by updating the state of one subcellular structure using random walk [60]. Identity swap move is to propose a new joint state by switching the states of two subcellular structures. Similarly, height swap move proposes a new joint state by exchanging only the z values of the states of two subcellular structures.

In the following subsections, we will present the proposal move function $q_m(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t})$ case by case, in which subscript m corresponds to the respective five move modalities aforementioned.

3.4.2.1 Disappear Move

For the disappear move, an object with identity number i is uniformly selected from the current identification number set n_t and its individual state is deleted from the joint state. The disappear move proposal is:

$$q_d(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = q_d(i) q_d(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}, i), \tag{3.15}$$

where $q_d(i) = 1/|n_t|$ is the proposal distribution for selecting i, $i \in \{n_t \setminus 0\}$, with $|\cdot|$ as the set cardinality operator, and $q_d(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}, i) = 1$. The deterministic mapping from $\mathbf{X}_{n_t}$ to $\mathbf{X}'_{n_t}$ is as $\{\ldots, \mathbf{X}'_{n_t.j} = \mathbf{X}_{n_t.j}, \ldots\}$, where $j \in \{n_t \setminus i\}$.

3.4.2.2 Appear Move

For the appear move, an object with identity number i is uniformly selected from set $\{\mathcal{A}_t \cup \bar{n}_t \setminus n_t\}$, with $\bar{n}_t = \cup_{k=1}^{N} n_{t-1}^{(k)}$, where N is the number of samples at time $t-1$ and $\mathcal{A}_t$ is the possible new object set at time t. $\mathcal{A}_t$ is constructed by image processing techniques and will be discussed in Section 3.6. The individual state of the identification number i is added to the joint state. The appear move proposal is:

$$q_a(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = q_a(i) q_a(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}, i), \tag{3.16}$$

with $q_a(i) = 1/|\mathcal{A}_t \cup \bar{n}_t \setminus n_t|$ as the proposal distribution for selecting i, $i \in \{\mathcal{A}_t \cup \bar{n}_t \setminus n_t\}$, $q_a(n'_t, \mathbf{X}'_{n_t};\ n_t, \mathbf{X}_{n_t}, i) = q_a(\mathbf{U})$, where $q_a(\mathbf{U})$ is the proposal of generating the state for the newly added object. The formulation of $q_a(\mathbf{U})$ will be presented in Section 3.6. The deterministic mapping relation from $(\mathbf{X}_{n_t}, \mathbf{U})$ to $\mathbf{X}'_{n_t}$ is as $\mathbf{X}'_{n_t.i} = \mathbf{U}$, $i \in \{\mathcal{A}_t \cup \bar{n}_t \setminus n_t\}$ and $\{\ldots, \mathbf{X}'_{n_t.j} = \mathbf{X}_{n_t.j}, \ldots\}$, $j \in n_t$.

3.4.2.3 Update Move

For the update move, an object identity number i is uniformly selected from the current identification number set n_t, and random walk is applied to it. The update move proposal is:

$$q_u(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = q_u(i) q_u(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}, i), \tag{3.17}$$

where $q_u(i) = 1/|n_t|$ is the proposal distribution for selecting i, $i \in \{n_t \backslash 0\}$, $q_u(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}, i) = q_u(\mathbf{U})$, where $q_u(\mathbf{U})$ is a Gaussian distribution. The deterministic mapping relation from $(\mathbf{X}_{n_t}, \mathbf{U})$ to $(\mathbf{X}'_{n_t}, \mathbf{U}')$ is as $\{\mathbf{U}' = -\mathbf{U}, \mathbf{X}'_{n_t.i} = \mathbf{X}_{n_t.i} + \mathbf{U}\}$, $i \in \{n_t \setminus 0\}$, and $\{\ldots, \mathbf{X}'_{n_t.j} = \mathbf{X}_{n_t.j}, \ldots\}$, $j \in \{n_t \setminus i\}$.

3.4.2.4 Identity Swap Move

For the identity swap move, two objects i and j in the current object set n_t are uniformly selected and their identities are exchanged. The identity swap move proposal is:

$$q_s(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = q_s(i, j) q_s(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}, i, j), \tag{3.18}$$

where $q_s(i, j) = 1/\binom{|n_t|}{2}$ is the proposal distribution for selecting the pair (i, j), and $q_s(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}, i, j) = 1$. The deterministic mapping from $\mathbf{X}_{n_t}$ to $\mathbf{X}'_{n_t}$ is as $\{\mathbf{X}'_{n_t.i} = \mathbf{X}_{n_t.j}, \mathbf{X}'_{n_t.j} = \mathbf{X}_{n_t.i}\}$, $i \in \{n_t \setminus 0\}$, $j \in \{n_t \setminus 0\}$, and $\{\ldots, \mathbf{X}'_{n_t.k} = \mathbf{X}_{n_t.k}, \ldots\}$, $k \in \{n_t \setminus i \setminus j\}$.

3.4.2.5 Height Swap Move

For the height swap move, using the bounding box $BB(\mathbf{X}_{n_t.i})$ information of each individual object state $\mathbf{X}_{n_t.i}$, we construct an overlap set O_{n_t} as:

$$O_{n_t} = \{(i, j) : i \in \{n_t \setminus 0\}, j \in \{n_t \setminus 0\}, BB(\mathbf{X}_{n_t.i}) \cap BB(\mathbf{X}_{n_t.j}) \neq \emptyset\}. \tag{3.19}$$

Then a pair of overlapped objects (i, j) is uniformly selected from O_{n_t} and the heights of the two objects are swapped by the z values of $\mathbf{X}_{n_t.i}$ and $\mathbf{X}_{n_t.j}$. The height swap move proposal is:

$$q_h(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = q_h(i, j) q_h(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}, i, j), \tag{3.20}$$

where $q_h(i, j) = 1/|O_{n_t}|$, and $q_h(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}, i, j) = 1$. The deterministic mapping from $\mathbf{X}_{n_t}$ to $\mathbf{X}'_{n_t}$ is $\{z'_{n_t.i} = z_{n_t.j}, z'_{n_t.j} = z_{n_t.i}, \xi'_{n_t.i} = \xi_{n_t.i}, \ \xi'_{n_t.j} = \xi_{n_t.j}\}$, $\xi \in \{l, w, x, y, \theta\}$, $(i, j) \in O_{n_t}$, and $\{\ldots, \mathbf{X}'_{n_t.k} = \mathbf{X}_{n_t.k}, \ldots\}$, where $k \in \{n_t \setminus i \setminus j\}$.

3.4.3 Acceptance Ratios

Applying the proposals of the different moves to equation (3.14), we get the following acceptance ratios α_d, α_a, α_u, α_s, and α_h for disappear move, appear move, update move, identity swap move, and height swap move, respectively:

$$\alpha_d(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = \frac{p(\mathbf{Z}_t|n'_t, \mathbf{X}'_{n_t}) p(n'_t, \mathbf{X}'_{n_t}|\mathbf{Z}_{1:t-1}) p_a q_a(i) q_a(\mathbf{U}')}{p(\mathbf{Z}_t|n_t, \mathbf{X}_{n_t}) p(n_t, \mathbf{X}_{n_t}|\mathbf{Z}_{1:t-1}) p_d q_d(i)}, \tag{3.21}$$

$$\alpha_a(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = \frac{p(\mathbf{Z}_t|n'_t, \mathbf{X}'_{n_t}) p(n'_t, \mathbf{X}'_{n_t}|\mathbf{Z}_{1:t-1}) p_d q_d(i)}{p(\mathbf{Z}_t|n_t, \mathbf{X}_{n_t}) p(n_t, \mathbf{X}_{n_t}|\mathbf{Z}_{1:t-1}) p_a q_a(i) q_a(\mathbf{U})}, \tag{3.22}$$

$$\alpha_u(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = \frac{p(\mathbf{Z}_t|n'_t, \mathbf{X}'_{n_t}) p(n'_t, \mathbf{X}'_{n_t}|\mathbf{Z}_{1:t-1})}{p(\mathbf{Z}_t|n_t, \mathbf{X}_{n_t}) p(n_t, \mathbf{X}_{n_t}|\mathbf{Z}_{1:t-1})}, \tag{3.23}$$

$$\alpha_s(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = \frac{p(\mathbf{Z}_t|n'_t, \mathbf{X}'_{n_t}) p(n'_t, \mathbf{X}'_{n_t}|\mathbf{Z}_{1:t-1})}{p(\mathbf{Z}_t|n_t, \mathbf{X}_{n_t}) p(n_t, \mathbf{X}_{n_t}|\mathbf{Z}_{1:t-1})}, \tag{3.24}$$

$$\alpha_h(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = \frac{p(\mathbf{Z}_t|n'_t, \mathbf{X}'_{n_t})}{p(\mathbf{Z}_t|n_t, \mathbf{X}_{n_t})}. \tag{3.25}$$

For the evaluation of $p(n_t, \mathbf{X}_{n_t}|\mathbf{Z}_{1:t-1})$, we use its mixture approximation as in equation (3.7):

$$\hat{p}(n_t, \mathbf{X}_{n_t}|\mathbf{Z}_{1:t-1}) = \frac{1}{N} \sum_{k=1}^{N} p(n_t, \mathbf{X}_{n_t}|n_{t-1}^{(k)}, \mathbf{X}_{n_{t-1}}^{(k)}), \tag{3.26}$$

where $p(n_t, \mathbf{X}_{n_t}|n_{t-1}^{(k)}, \mathbf{X}_{n_{t-1}}^{(k)})$ is evaluated by using equation (3.11) and equation (3.12) as:

$$p(n_t, \mathbf{X}_{n_t}|n_{t-1}^{(k)}, \mathbf{X}_{n_{t-1}}^{(k)}) = p(n_t|n_{t-1}^{(k)}, \mathbf{X}_{n_{t-1}}^{(k)}) \prod_{j \in \mathcal{B}_t} p(\mathbf{X}_{n_t.j}) \prod_{i \in \mathcal{S}_t} p(\mathbf{X}_{n_t.i}|\mathbf{X}_{(n_{t-1}).i}^{(k)}), \tag{3.27}$$

with $\mathcal{B}_t = n_t \setminus n_{t-1}^{(k)}$, $\mathcal{S}_t = n_t \cap n_{t-1}^{(k)}$ as previously introduced in section 3.3. There are two cases for object identification number j in $\mathcal{B}_t$, $j \in \mathcal{A}_t$ and $j \notin \mathcal{A}_t$. $\mathcal{A}_t$ is the possible new object set at time t as mentioned in appear move. For the first case, $p(\mathbf{X}_{n_t.j}) = p_{new}$, with $j \in \mathcal{A}_t$. For the second case, $p(\mathbf{X}_{n_t.j})$ is defined to be $p(\mathbf{X}_{n_t.j}) \triangleq \sum_{i \in N_{t-1}^{(j)}} p(\mathbf{X}_{n_t.j}|\mathbf{X}_{n_{t-1}.j}^{(i)})/|N_{t-1}^{(j)}|$, with $N_{t-1}^{(j)} = \{i : j \in n_{t-1}^{(i)}\}$, the set of samples at time $t-1$ containing object j. $n_{t-1}^{(i)}$ is the object set for particle i at time $t-1$. $p(n_t|n_{t-1}^{(k)}, \mathbf{X}_{n_{t-1}}^{(k)})$ is defined to be equivalent for all n_t, $n_{t-1}^{(k)}$, and $\mathbf{X}_{n_{t-1}}^{(k)}$, since we assume the equal chance of each object set n_t.

3.5 2D Subcellular Structure Detection and Representation

One of the difficulties in tracking multiple caveolar structures is how to detect and represent the objects in the image scene. Although for common GFP image sequence, a thresholding method can easily distinguish the foreground from the background, it suffers from being incapable of separating objects in the foreground. As shown in Fig. 3.2(a), the object with a white boundary in the left portion and the object with a yellow boundary in the middle are under-segmented. Another widely used segmentation method in biological image applications [61] is watershed. Figure 3.2(b) displays the result that has looped segmented objects, which is difficult for multiple object tracking. The result of direct allocating regional maxima on the original image is illustrated in Fig. 3.2(c). However, we can see an over-labeled number of objects by regional maxima even after the image was first filtered by a low pass filter.

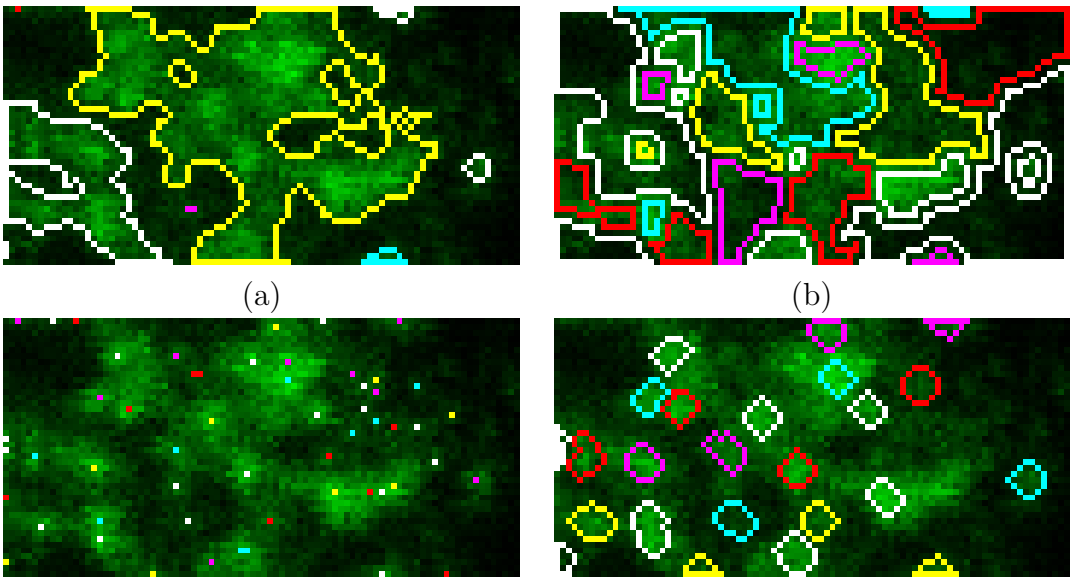


Figure 3.2. Comparison of segmentation methods. (a), (b), (c), and (d) are original images superimposed by segmentation results by thresholding, watershed method, regional maxima, and marker image, respectively. Different objects are indicated by different color boundaries.

To achieve a reliable segmentation, our segmentation algorithm starts with a morphological reconstructive opening for object detection [62]. Regional maxima is used to represent each object. The idea is similar to track a human being movement by only tracking the face without using the whole body information, since the face information is enough to distinguish each person in most cases. For an image at time t with intensity as I_t and a structuring element as S, the gray-scale reconstructive opening operation $\circledcirc$ is represented as:

$$I_t \circledcirc_C S = I_t \odot_C (I_t \circ S), \tag{3.28}$$

where $\odot$ is the gray scale reconstruction operation, $\circ$ is the gray scale morphological opening, and C is the connectivity definition for $\odot$. We use the regional maxima of $I_t \circledcirc_C S$ to identify each object.

Nonetheless, the regional maxima of $I_t \circledcirc_C S$ is unable to provide enough feature information for object matching during the tracking process. Therefore, we apply the morphological dilation operation on each regional maximum to include more feature information under the condition that the dilated objects do not overlap with each other. Since the size of objects varies a lot, from medium (about 100 pixels) to small (about 20 pixels), we apply a smallest structure element $[0, 1, 0; 1, 1, 1; 0, 1, 0]$ for the dilation operation, which is iso-tropic in horizontal and vertical directions. The corresponding binary image at time t is called *marker image*, denoted as M_t. The dilated version of regional maxima superimposed on the real image is shown in Fig. 3.2(d).

The benefit of using dilated regional maxima of $I_t \circledcirc_C S$ to detect object comes from the fact that it can provide well labeled object results for close-neighbored objects and is more robust in the whole image sequence. Here, we emphasize more on the object identification and characterization ability of the dilated regional maxima for tracking purpose, not its capability to produce accurate object segmentation boundary, though this can be refined by post-processing.

Once the marker image is obtained, we use the minimum bounding box (MBB) to represent each marker. Suppose a single pixel in one subcellular object is represented as a 2D vector $\boldsymbol{u} = (x, y)^T$, where x and y are the coordinates in the 2D plane and T is the transpose operation, the center position vector $\bar{\boldsymbol{u}} = (\bar{x}, \bar{y})^T$ of the region can be calculated as:

$$\bar{\boldsymbol{u}} = \frac{1}{L} \sum_{i=1}^{L} \boldsymbol{u}_i, \tag{3.29}$$

where L is the number of pixels in the region. The covariance matrix of the position vector $\boldsymbol{u}$ for the object region can be obtained by:

$$\Sigma_p = \frac{1}{L} \sum_{i=1}^{L} \boldsymbol{u}_i \boldsymbol{u}_i^T - \bar{\boldsymbol{u}} \bar{\boldsymbol{u}}^T. \tag{3.30}$$

The direction of the major eigenvector v_1 of covariance matrix Σ_p determines the orientation angle θ in object state $\mathbf{X}_{t.i}$. The largest lengths of segmented object in the directions of major eigenvector v_1 and minor eigenvector v_2 decide length l and width w in $\mathbf{X}_{t.i}$. An illustrative graph is provided in Figure 3.3.

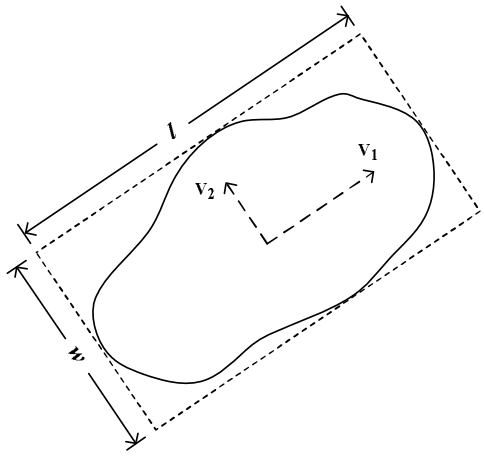


Figure 3.3. Segmented object representation. The region of interest is delimited by the solid curve line and is represented using a dotted rectangle, object center, and major eigenvector.

The relationship between regional maxima of $I_t \circledcirc_C S$, markers and their MBBs are shown in Fig 3.4. As can be seen from the superimposed images, more pixel information is added to MBB representation of each object while still keep its identity separated.

3.6 2D Marker Residual Image-Guided Appearance Model

Unlike other multiple object tracking methods, in which objects appear/disappear only in certain fixed regions [63], [51], [52], in our data sets, objects can appear/disappear in any part of the cytoplasm. This poses a big challenge for the varying object num-

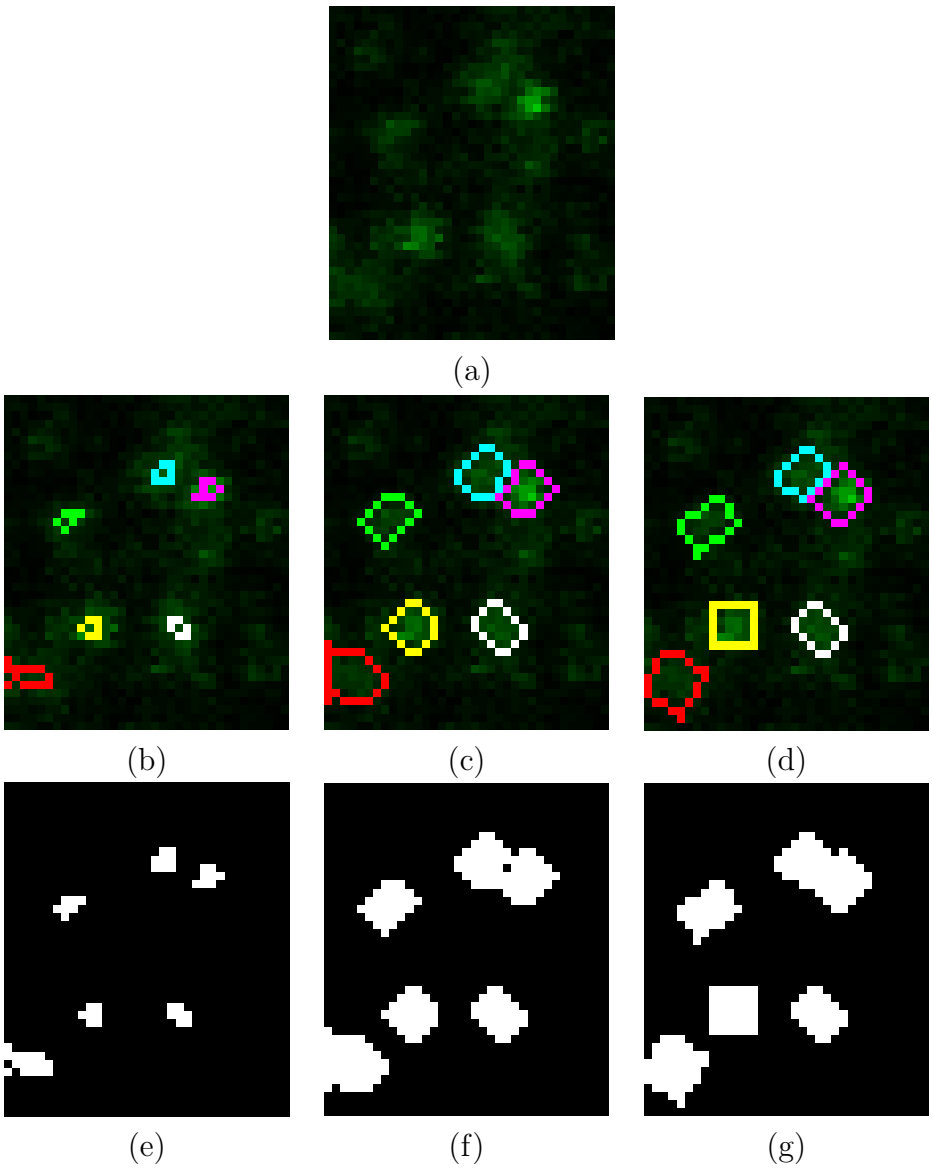


Figure 3.4. From regional maxima image to state. (a) is the original image. (e) is regional maxima image. (f) is marker image. (g) is the MBB image of marker image. (b), (c), and (d) are the original images superimposed by (e), (f), and (g), respectively. Different regional maxima, markers, and MBBs are indicated by different color boundaries in the superimposed image.

ber tracking problem. Here we introduce a marker residual image in order to solve the problem. For a marker image M_t with n markers, it can be represented as:

$$M_t = \cup_{i=1}^{n} M_{t.i}. \tag{3.31}$$

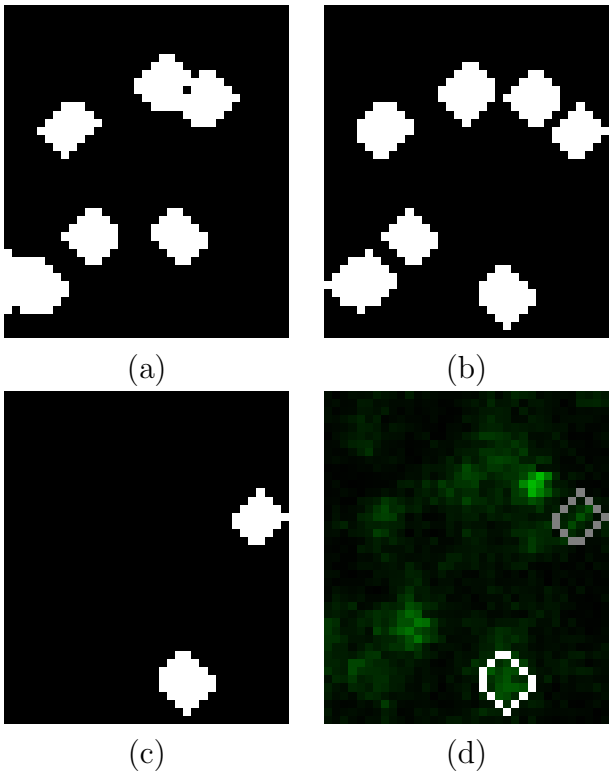


Figure 3.5. Marker residual image. (a) Marker image of frame 61. (b) Marker image of frame 62. (c) Marker residual image of frame 62. (d) Original image superimposed by marker residual image, with different markers indicated by different color boundaries.

where $M_{t,i}$ is a binary image with segmented object i. A new image called marker residual image $\mathbb{M}_t$ that depicts the newly appearing objects is defined as:

$$\mathbb{M}_t = \{M_{t.i} : M_{t.i} \cap M_{t-1} = \emptyset, i = 1, \cdots, n\}. \tag{3.32}$$

As illustrated in Fig. 3.5, marker images M_{61} and M_{62} are shown in Figs. 3.5(a) and (b), respectively. From the residual image $\mathbb{M}_{62}$, the two newly detected object regions can be seen in Fig. 3.5(c).

From the newly detected n objects, a set of binary images $\mathbb{M}_{t.i}, (i = 1, \cdots, n)$, is available and the object set is denoted as $\mathcal{A}_t$. The previously introduced proposal function $q_a(\mathbf{U})$ for the new object is defined as follows:

$$q_a(\mathbf{U}) = \begin{cases} p(\mathbf{X}_{n_t.j}|\mathbb{M}_{t.j}), & \text{if } j \in \mathcal{A}_t \\ \sum_{i \in N_{t-1}^{(j)}} P(\mathbf{X}_{n_t.j}|\mathbf{X}_{(n_{t-1}).j}^{(i)})/|N_{t-1}^{(j)}|, & \text{if } j \notin \mathcal{A}_t, \end{cases} \tag{3.33}$$

where $p(\mathbf{X}_{n_t.j}|\mathbb{M}_{t.j})$ generates the state (l, w, θ, x, y) by a Gaussian distribution centered around the object in the image $\mathbb{M}_{t.j}$, $N_{t-1}^{(j)} = \{i : j \in n_{t-1}^{(i)}\}$ is the set of samples at time $t-1$ containing object j.

3.7 2D Observation Model

The joint observation model is defined as:

$$p(\mathbf{Z}_t|\mathbf{X}_t) = \prod_{i \in n_t} p(\mathbf{Z}_{t.i}|\mathbf{X}_{t.i}). \tag{3.34}$$

For the measurement of each individual object state $\mathbf{X}_{t.i}$, we use the amount and intensity profile of the pixels inside its bounding box to match with its reference $R_{t.i}$, which is obtained at the beginning of each object trajectory. Three properties between $\mathbf{X}_{t.i}$ and $R_{t.i}$ are considered in our observation measurement: (1) area; (2) intensity mean; (3) sorted sum-of-absolute-differences (SSAD). The intuition behind the first one is that although the shape of the subcellular structure is unstable, its size is relatively stable. The intensity mean and SSAD are used to characterize the intensity profile of pixels inside the state bounding box, with intensity mean for the global similarity and SSAD for the similarity in detail. Thus we define the observation model for each individual object as:

$$p(\mathbf{Z}_{t.i}|\mathbf{X}_{t.i}) = p(\mathbf{Z}_{t.i}^a|\mathbf{X}_{t.i})p(\mathbf{Z}_{t.i}^m|\mathbf{X}_{t.i})p(\mathbf{Z}_{t.i}^d|\mathbf{X}_{t.i}), \tag{3.35}$$

with $p(\mathbf{Z}^a_{t.i}|\mathbf{X}_{t.i})$, $p(\mathbf{Z}^m_{t.i}|\mathbf{X}_{t.i})$, and $p(\mathbf{Z}^d_{t.i}|\mathbf{X}_{t.i})$, for area measurement, intensity mean, and SSAD, respectively.

Suppose the set of intensity of pixels inside the bounding box of $\mathbf{X}_{t.i}$ is $A_n = \{a_1, \ldots, a_j, \ldots, a_n\}$, where n is the amount of pixels and a_j stands for the intensity of the individual pixel. The mean intensity of A_n is:

$$I_{A_n} = \sum_{j=1}^{n} a_j/n. \tag{3.36}$$

Same to $R_{t.i}$, we have its intensity set $B_m = \{b_1, \ldots, b_j, \ldots, b_m\}$ and intensity mean $I_{B_m} = \sum_{j=1}^{m} b_j/m$, with m as the amount of pixels.

We define $p(\mathbf{Z}^a_{t.i}|\mathbf{X}_{t.i})$ as:

$$p(\mathbf{Z}^a_{t.i}|\mathbf{X}_{t.i}) \propto \exp\left(-\lambda_a \left(1 - \frac{\min(n,m)}{\max(n,m)}\right)^2\right). \tag{3.37}$$

Similarly, $p(\mathbf{Z}^m_{t.i}|\mathbf{X}_{t.i})$ is defined by:

$$p(\mathbf{Z}^m_{t.i}|\mathbf{X}_{t.i}) \propto \exp\left(-\lambda_m \left(1 - \frac{\min(I_{A_n}, I_{B_m})}{\max(I_{A_n}, I_{B_m})}\right)^2\right). \tag{3.38}$$

Here, λ_a, λ_m are hyperparameters.

We define the SSAD between two regions as the intensity sets matching problem. Supposing there are two equal-sized objects A_n and B_n, where n stands for the number of pixels in the object, we use the minimum sum-of-absolute-differences (SAD) between A_n and B_n as the distance measurement between them. SAD has been used a lot in image similarity comparison problem. Nonetheless, it is limited to template-type matching, where the pixel correspondences between two images are already established. Unfortunately, the template-type matching does not work in our case, since the object is deforming and its intensity is not stable at any specific part. Thus we have to find the minimum SAD between A_n and B_n by enumerating $n!$ possible cases. The computation cost is high in tracking case. Here we propose

one method saving the computation cost from weak lower bound $\omega(2^n)$ to tight upper bound $O(n \lg n)$ by sorting the pixels in A_n and B_n in descending order to get A'_n and B'_n and let the one-to-one correspondence pair as $a'_i \sim b'_i$, then the minimum SAD_{min} between A_n and B_n is as:

$$SAD_{min}(A_n, B_n) = \sum_{i}^{n} |a'_i - b'_i)|. \tag{3.39}$$

Please see Appendix B for our proof.

To handle the mismatched size of A_n and B_m, without loss of generality, let $m < n$, we use the first m high intensity pixels in each set for the computation of SAD_{min}. The intuition is that in LSCM images, objects are discerned and characterized by their illuminance intensity. The final intensity similarity between A_n and B_m is defined by

$$SSAD(A_n, B_m) = \frac{1}{\min\{n, m\}} \sum_{i}^{\min\{n,m\}} |a'_i - b'_i)|, \tag{3.40}$$

Finally, $p(\mathbf{Z}^d_{t.i}|\mathbf{X}_{t.i})$ is as:

$$p(\mathbf{Z}^d_{t.i}|\mathbf{X}_{t.i}) \propto \exp\left(-\lambda_d \cdot (SSAD(A_n, B_m))^2\right), \tag{3.41}$$

with λ_d as a hyperparameter too. The intuition of SSAD is that for the small subcellular structure, its salient points are the pixels with high intensity value. By comparing the intensity values of these salient points, we can get the information about how similar two small structures are, since there is no obvious spatial pattern of the pixels within such small areas.

In the overlapping situation when the spatial support of $\mathbf{X}_{t.i}$ is occluded by other objects, we define an Euclidean transformation T (a composition of translation and rotation), which translates the center of $R_{t.i}$ to the center of $\mathbf{X}_{t.i}$ and rotates it to have the same orientation as $\mathbf{X}_{t.i}$, as shown in Fig. 3.6. Let $R'_{t.i} = T(R_{t.i})$. We use the un-occluded spatial parts of $\mathbf{X}_{t.i}$ and $R'_{t.i}$ for intensity mean and SSAD calculation.

In the case that at least one of $\mathbf{X}_{t.i}$ and $R'_{t.i}$ is fully overlapped by the upper objects, $p(\mathbf{Z}_{t.i}|\mathbf{X}_{t.i})$ can not be measured. We use the best guess for the observation, which is the mean of the observations of all the samples of object i at previous time $t-1$.

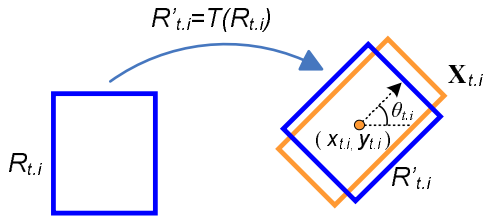


Figure 3.6.Illustration of the Euclidean transformation when overlapping.

3.8 Summary of the Algorithm for Multiple 2D Subcellular Structure Tracking

The algorithm for multiple subcellular structure tracking is summarized as follows:

1. Automatically detect the subcellular structures using marker residual image. Estimate the state $\mathbf{X}_{0.i}$ of each individual subcellular structure, and combine them into $\mathbf{X}_0$. Sample the joint state $\mathbf{X}_0$ with N samples. Thus the initial distribution of $\mathbf{X}_0$ is approximated by $p(\mathbf{X}_0) \approx \{\mathbf{X}_0^{(s)}\}$, $s = 1, \ldots, N$. Set $t = 0$.
2. Set $t = t+1$. Draw a sample of $\mathbf{X}_t$ from the prediction density $\sum_{s=1}^{N} p(\mathbf{X}_t|\mathbf{X}_{t-1}^{(s)})/N$.
3. Apply RJMCMC method to draw N samples.
   (a) Draw a sample w from the uniform distribution on $(0, 1)$.
   (b) If $0 \leq w < p_d$, apply disappear move. Delete the state of one subcellular structure from the joint state using equation (3.15).
   (c) If $p_d \leq w < p_d + p_a$, apply appear move. Add the state of one subcellular structure to the joint state using equation (3.16).

(d) If $p_d + p_a \leq w < p_d + p_a + p_u$, apply update move. Update the state of one subcellular structure in the joint state using equation (3.17).

(e) If $p_d + p_a + p_u \leq w < p_d + p_a + p_u + p_s$, apply identity swap move. Swap the states of two subcellular structures in the joint state using equation (3.18).

(f) If $p_d + p_a + p_u + p_s \leq w$, apply height swap move. Swap the heights of two subcellular structures in the joint state using equation (3.20).

(g) Calculate the acceptance ratio $\alpha(\mathbf{X}_t, \mathbf{X}'_t)$ based on the move type using equations (3.21), (3.22), (3.23), (3.24), and (3.25).

(h) Draw a sample w' from the uniform distribution on $(0, 1)$. If $w' < \alpha(\mathbf{X}_t, \mathbf{X}'_t)$, use $\mathbf{X}'_t$ as the new joint state; else use $\mathbf{X}_t$. Add the new state to the final sample set.

4. Go to step 2.

3.9 Experimental Results

To investigate the performance of our proposed methods, we demonstrate the experimental results using both synthetic images and confocal microscopy images.

3.9.1 Synthetic Image Sequence Tracking

The synthetic image sequence is of size 80×80 and has nine frames with added Gaussian noise (Figure 3.7). It contains the phenomena of object overlapping and disappearing. The four objects in the sequence are O1, O2, O3, and O4 with size 10×5, 14×7, 9×6, and 7×7, respectively. In the sequence, objects O1 and O2 execute translational motion with corresponding constant vertical and horizontal speeds as $(2, 2)$ and $(-2, -2)$, respectively, and overlap with each other. Objects O3 and O4 perform Brownian motion, and object O3 disappears in Frames 8 and

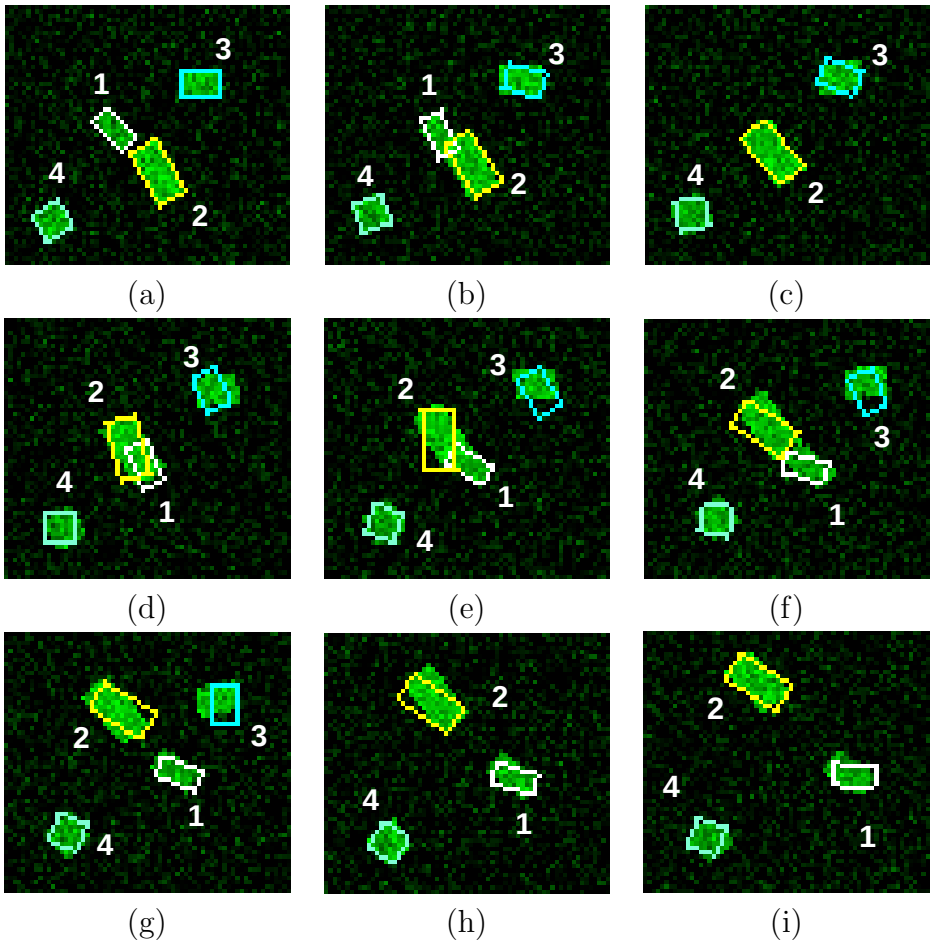


Figure 3.7. Tracking results of synthetic image sequence. (a) to (i) are image frames with number 1 to 9.

9. A clustering method based on expectation maximization (EM) algorithm [64] was applied to separate the background and foreground.

As can be seen in Figure 3.7(b), our method detected the overlapping event between objects O1 and O2, and continued to track them till the end of the image sequence even after the full occlusion of object O1 in frames 3 and 4. The tracking result reveals that object merging (frames 3 and 4) and splitting (frames 6 and 7), though not explicitly modeled in the algorithm, can be handled successfully. The disappearing of object O3 in frames 8 and 9 was also detected correctly.

For the parameter initialization, the state transition density of (l, w, θ, x, y) for the synthetic data is set to be a Gaussian distribution with zero mean vector and covariance matrix $\Sigma = [1, 0, 0, 0, 0;\ 0, 1, 0, 0, 0;\ 0, 0, 0.1, 0, 0;\ 0, 0, 0, 30, 0;\ 0, 0, 0, 0, 30]$. The random walk covariance matrix is chosen as $\Sigma = [0.1, 0, 0, 0, 0;\ 0, 0.1, 0, 0, 0;\ 0, 0, 0.01, 0, 0;\ 0, 0, 0, 9, 0;\ 0, 0, 0, 0, 9]$. For the augmented variable z, $p(z_{t.i}|z_{(t-1).i})$ is set to be a uniform distribution on $(0, 1)$. p_d, p_a, p_u, p_s, and p_h are heuristically set to be 0.1, 0.1, 0.6, 0.1, and 0.1, respectively. $\lambda_a = 7$, $\lambda_m = 5$, and $\lambda_d = 0.4$. For RJMCMC sample generation, sample number $N = 800$ was used in the experiment. p_{new} is set to be 0.005. The covariance matrix of (l, w, θ, x, y) for the Gaussian distribution $p(\mathbf{X}_{n_t.j}|\mathbb{M}_{t.j})$ is set as $\Sigma = [1, 0, 0, 0, 0;\ 0, 1, 0, 0, 0;\ 0, 0, 0.1, 0, 0;\ 0, 0, 0, 2, 0;\ 0, 0, 0, 0, 2]$.

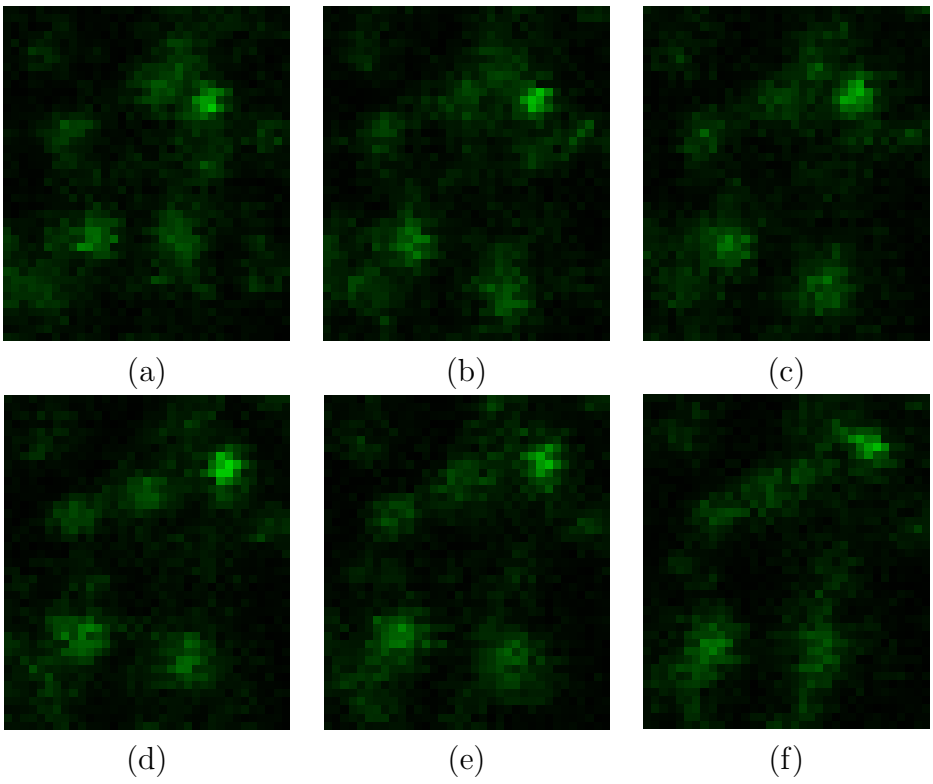


Figure 3.8. Original real image sequence one. (a) to (f) are image frames at time 0 sec, 1.6 sec, 3.2 sec, 4.8 sec, 6.4 sec, and 8.0 sec.

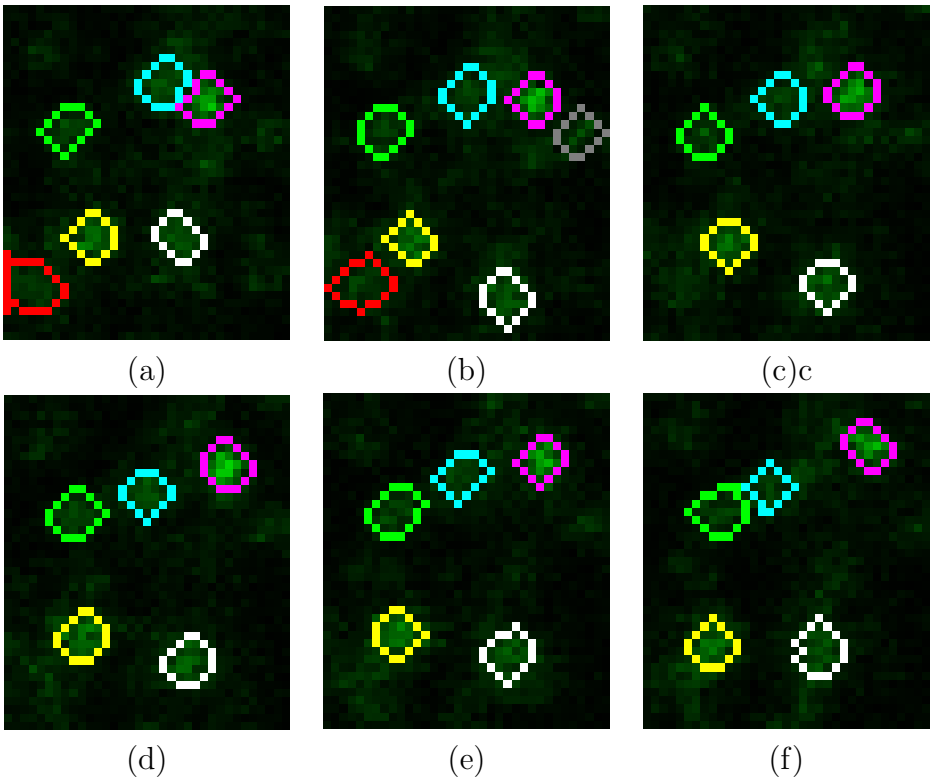


Figure 3.9. Results of marker image superimposed on the real image sequence, with different markers indicated by different color boundaries. (a) to (f) are image frames at time 0 sec, 1.6 sec, 3.2 sec, 4.8 sec, 6.4 sec, and 8.0 sec.

3.9.2 Real Image Sequence Tracking

An image sequence by Leica TCS-SP laser scanning confocal microscope was obtained to study of the trafficking of caveolar membranes on microtubules and the actin cytoskeleton. Caveolae are plasma membrane specializations that mediate the internalization of extracellular ligands, including cholera toxin B and some viruses. Caveolin 1 is a marker protein for caveolar membranes. Time lapse sequences of cells expressing caveolin 1-GFP were taken with a Leica TCS-SP laser scanning confocal microscope with a 100x objective lens. The video images are of size 512×512 with a time interval of 1.6 sec between two frames. We observed the average size of tagged structures is less than 10×10 pixels and appear/disappear, split and merge as time

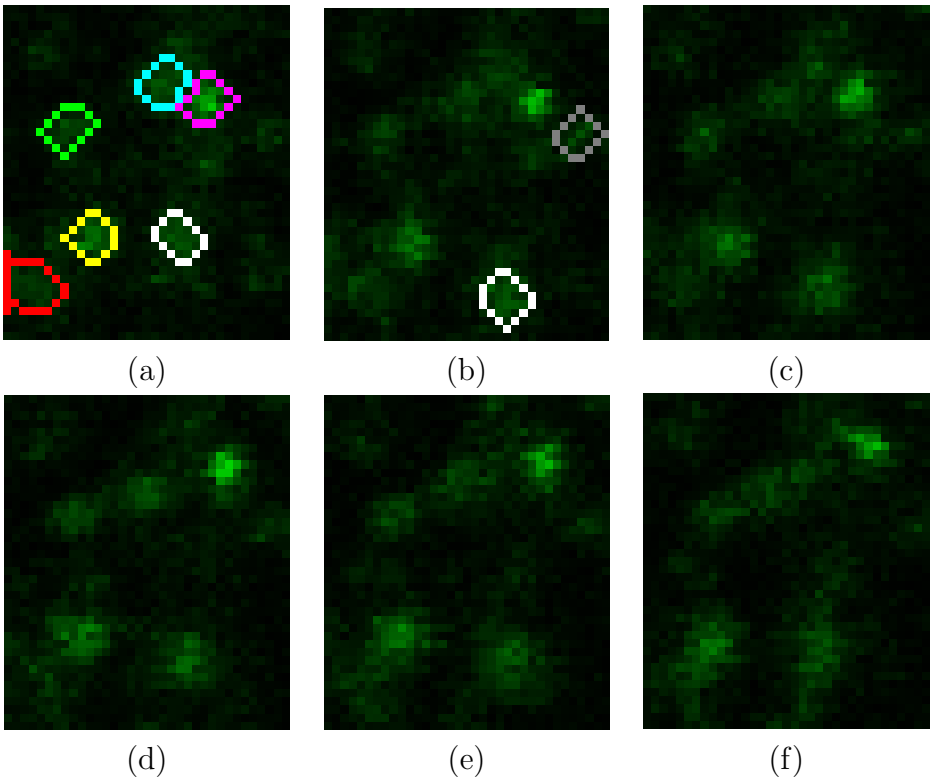


Figure 3.10. Results of marker residual image superimposed on real image sequence one, with different residual markers indicated by different color boundaries. (a) to (f) are image frames at time 0 sec, 1.6 sec, 3.2 sec, 4.8 sec, 6.4 sec, and 8.0 sec.

goes on, which indicates challenges caused by small size, homogeneous regions, and change of motion modality. Tracking results of two types of motion modalities, Brownian and translational motion, will be presented. Similar as before, image background will be separated using an expectation maximization (EM) method.

3.9.2.1 Brownian Motion

As an example of Brownian motion tracking, image sequences of a small portion of the image scene, from 0 second to 8.0 second, are shown in Figure 3.8. The marker images superimposed on the original images at different times are shown in Figure 3.9, with related marker residual images shown in Figure 3.10. The final tracking results

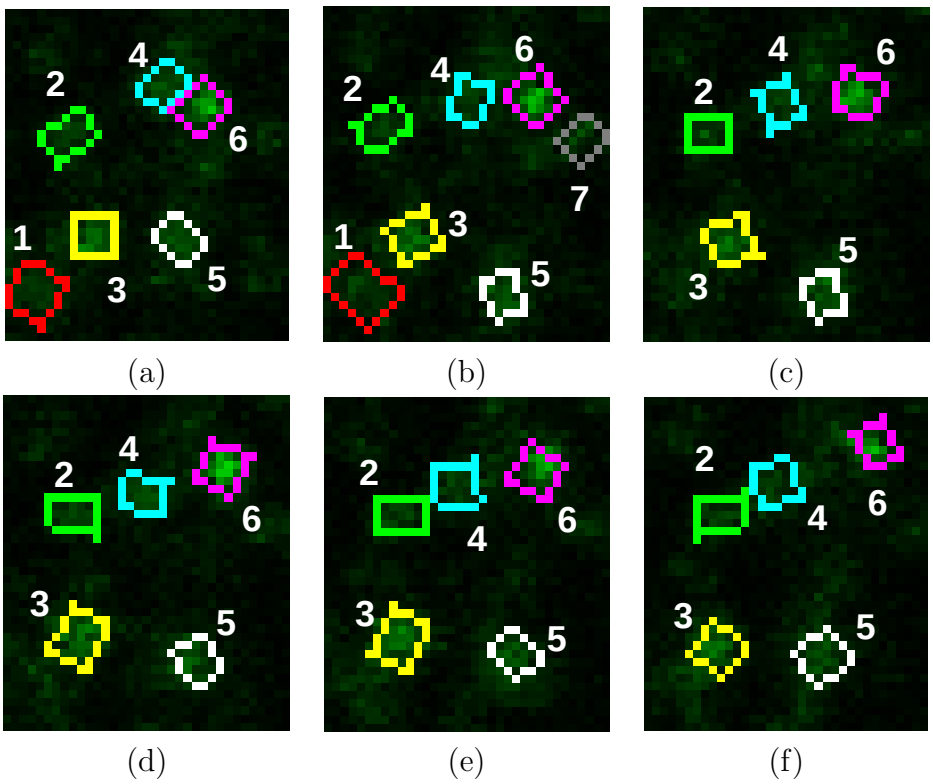


Figure 3.11. Tracking result confocal image sequence one. (a) to (f) are image frames at time 0 sec, 1.6 sec, 3.2 sec, 4.8 sec, 6.4 sec, and 8.0 sec.

are shown in Figure 3.11, where the tracked subcellular structures are depicted as rectangular bounding boxes.

Initially there are six objects detected at 0 second. Then object O7 appears at 1.6 seconds. At 3.2, 4.8, and 6.4, and 8.0 seconds, both objects O1 and O7 disappear. In the whole process, objects O2, O3, O4, and O6 show constrained Brownian motion. Object O5 demonstrates unconstrained Brownian motion with a large displacement. Notice that at 1.6 sec, there are two possible new objects detected. As shown in Figure 3.10(b), one is delimited by white boundary and the other is delimited by gray boundary. Our algorithm correctly differentiates that the one with gray boundary is a true new object, which is detected as object O7 in Figure 3.11(b). The one with

white boundary is ended up being the matching target for object O5 that is already alive at 0 second.

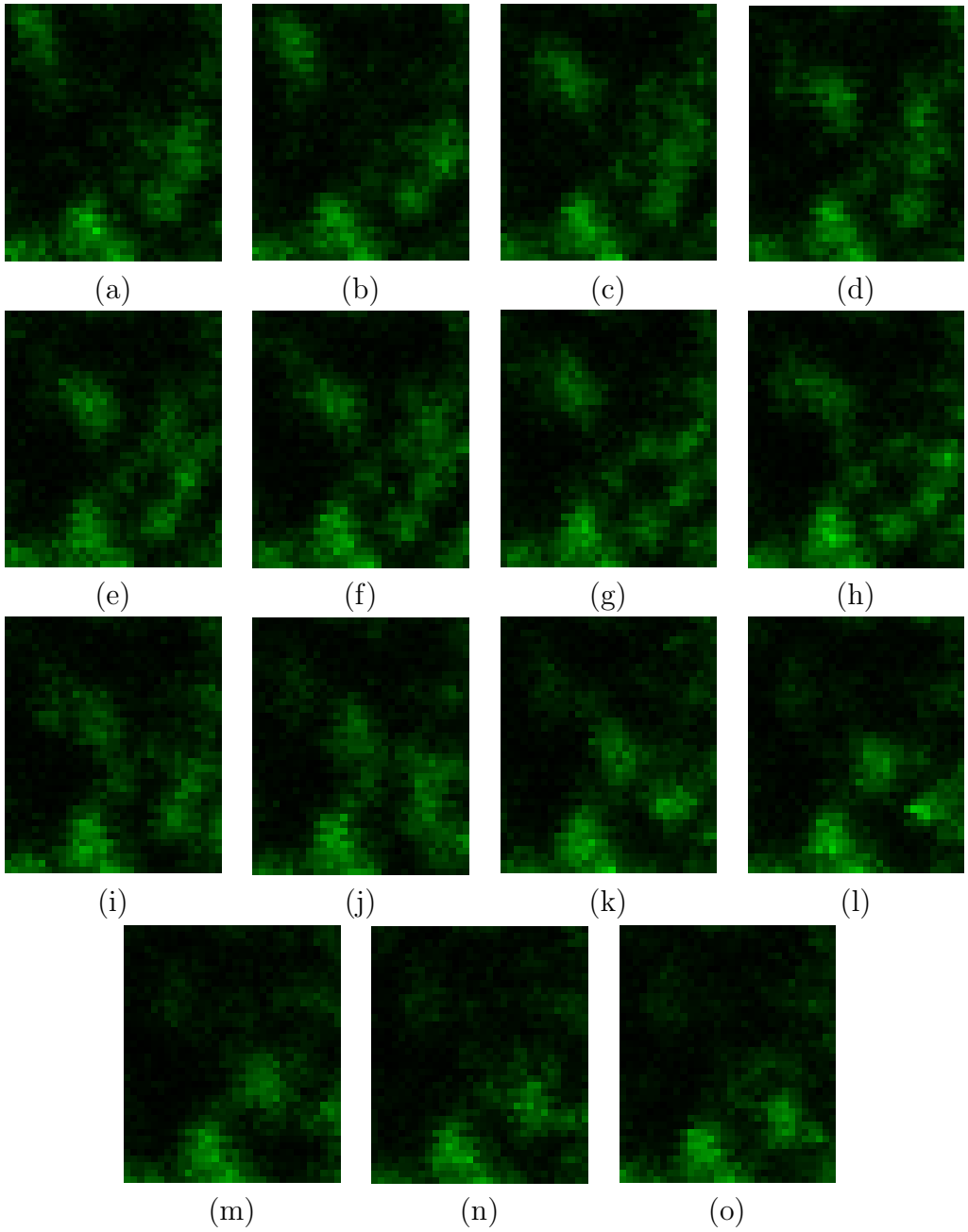


Figure 3.12. Original real image sequence two. (a) to (o) are image frames at time 0 sec, 1.6 sec, 3.2 sec, 4.8 sec, 6.4 sec, 8.0 sec, 9.6 sec, 11.2 sec, 12.8 sec, 14.4 sec, 16.0 sec, 17.6 sec, 19.2 sec, 20.8 sec, and 22.4 sec.

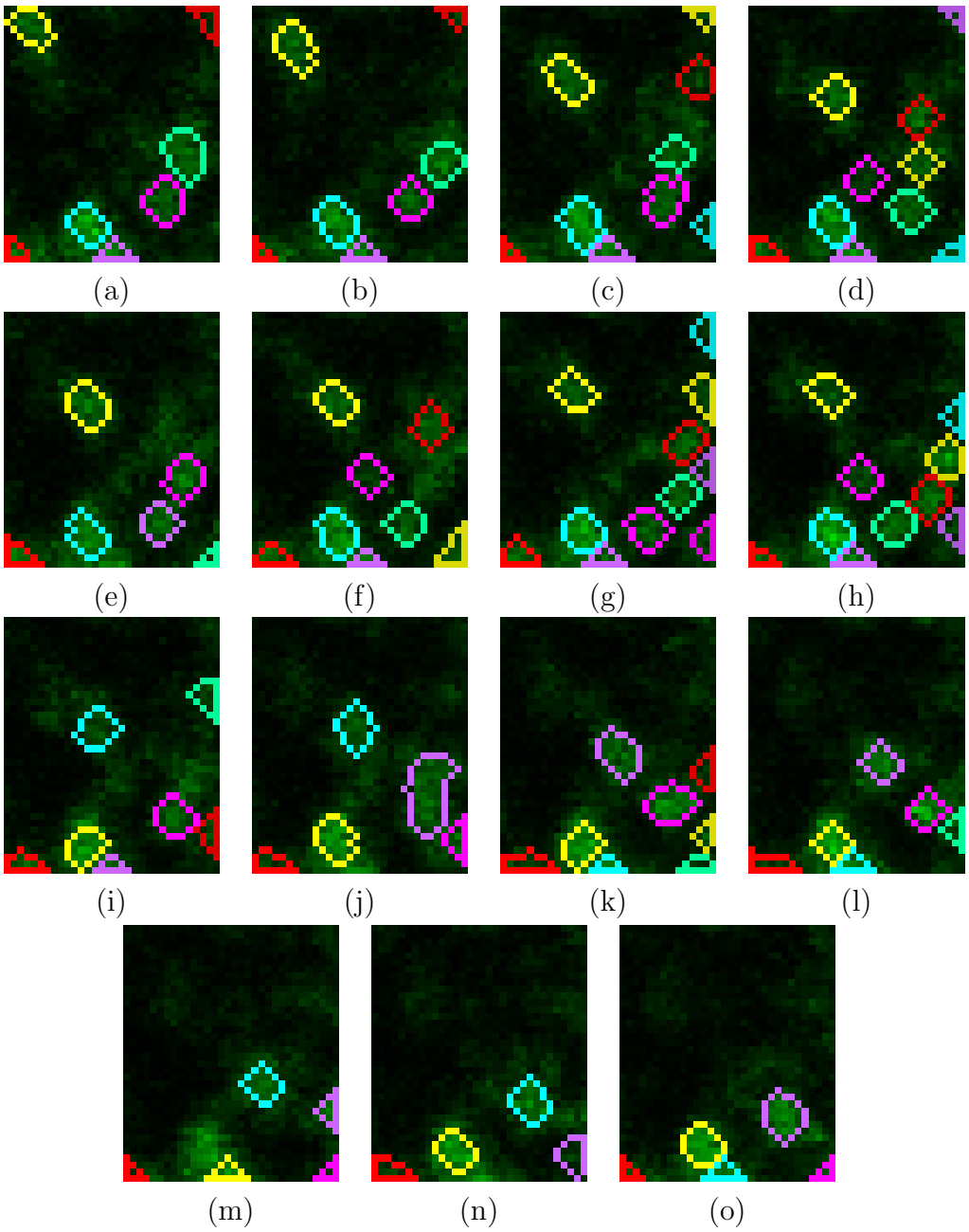


Figure 3.13. Marker image superimposed on the original image sequence two. Different markers are delimitated by different color boundaries. (a) to (o) are image frames at time 0 sec, 1.6 sec, 3.2 sec, 4.8 sec, 6.4 sec, 8.0 sec, 9.6 sec, 11.2 sec, 12.8 sec, 14.4 sec, 16.0 sec, 17.6 sec, 19.2 sec, 20.8 sec, and 22.4 sec.

3.9.2.2 Translational Motion

The sequence we present here shows caveolin trafficking displaying both short-range saltatory movements and long-range microtubule-based movements that these cells exhibit. Data analysis results for one video sequence are presented here. Its recording time span is 22.4 second with a 1.6 second interval between frames. A region of 38×32 pixels out of the original 512×512 image is used to demonstrate the tracking result. Several of the typical motion patterns (appear, disappear, re-appear, translation, and Brownian) were observed within the selected window.

The original images are shown in Fig. 3.12. The corresponding marker images and marker residual images superimposed on original images are shown in Fig. 3.13 and Fig. 3.14, respectively. 24 new objects are detected from the resulting marker residual images. After the RJMCMC move detection, 12 of them are selected. The tracking result for these 12 objects are shown in Fig. 3.15 where the same object is depicted by the same color. These objects exhibit either translational (objects O2 and O8), or Brownian (objects O1, O4, O7, O9, O10), or a combination of both (objects O3, O5, O6, O11, O12). As an obvious example, object O2 executes translational motion, moving from the upper left corner to the lower right corner. Figure 3.16 displays the tracking trajectories of these objects over the 22.4 second recording time.

To quantitatively evaluate the tracking performance, we compared the results with ground truth that is obtained by a human-in-the-loop segmentation editor. The average displacement error between the center of the tracked subcellular structure and the center of the ground truth was calculated for each video frame at 1.6 second recording interval (Fig. 3.17). Fig. 3.18 shows the change of number of objects during tracking process. "Truth" means the real number of objects. "Correct" tracked objects means that an object is in the ground truth and is tracked by our method.

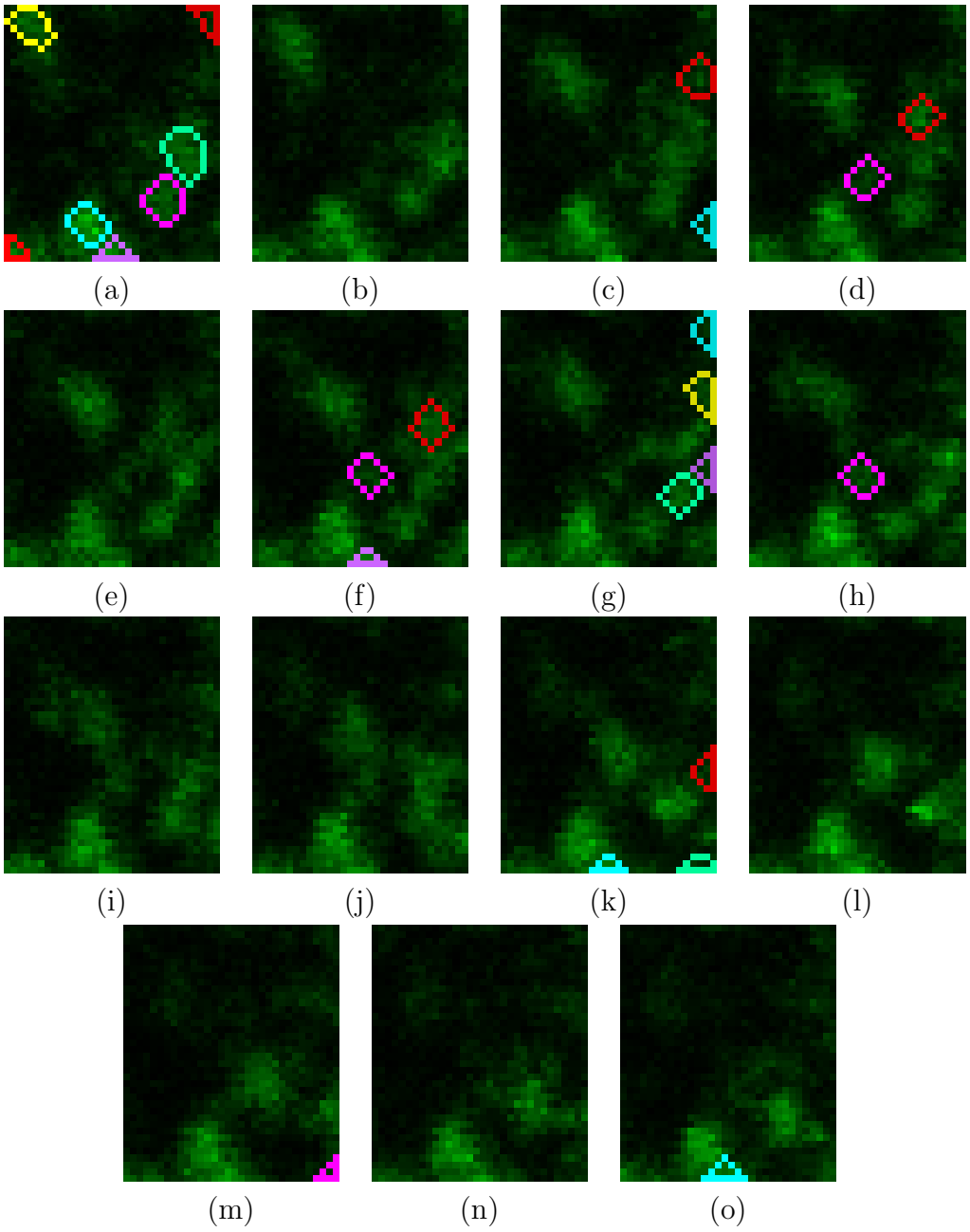


Figure 3.14. Marker residual image superimposed on the original image sequence two. Different marker residuals are delimitated by different color boundaries. (a) to (o) are image frames at time 0 sec, 1.6 sec, 3.2 sec, 4.8 sec, 6.4 sec, 8.0 sec, 9.6 sec, 11.2 sec, 12.8 sec, 14.4 sec, 16.0 sec, 17.6 sec, 19.2 sec, 20.8 sec, and 22.4 sec.

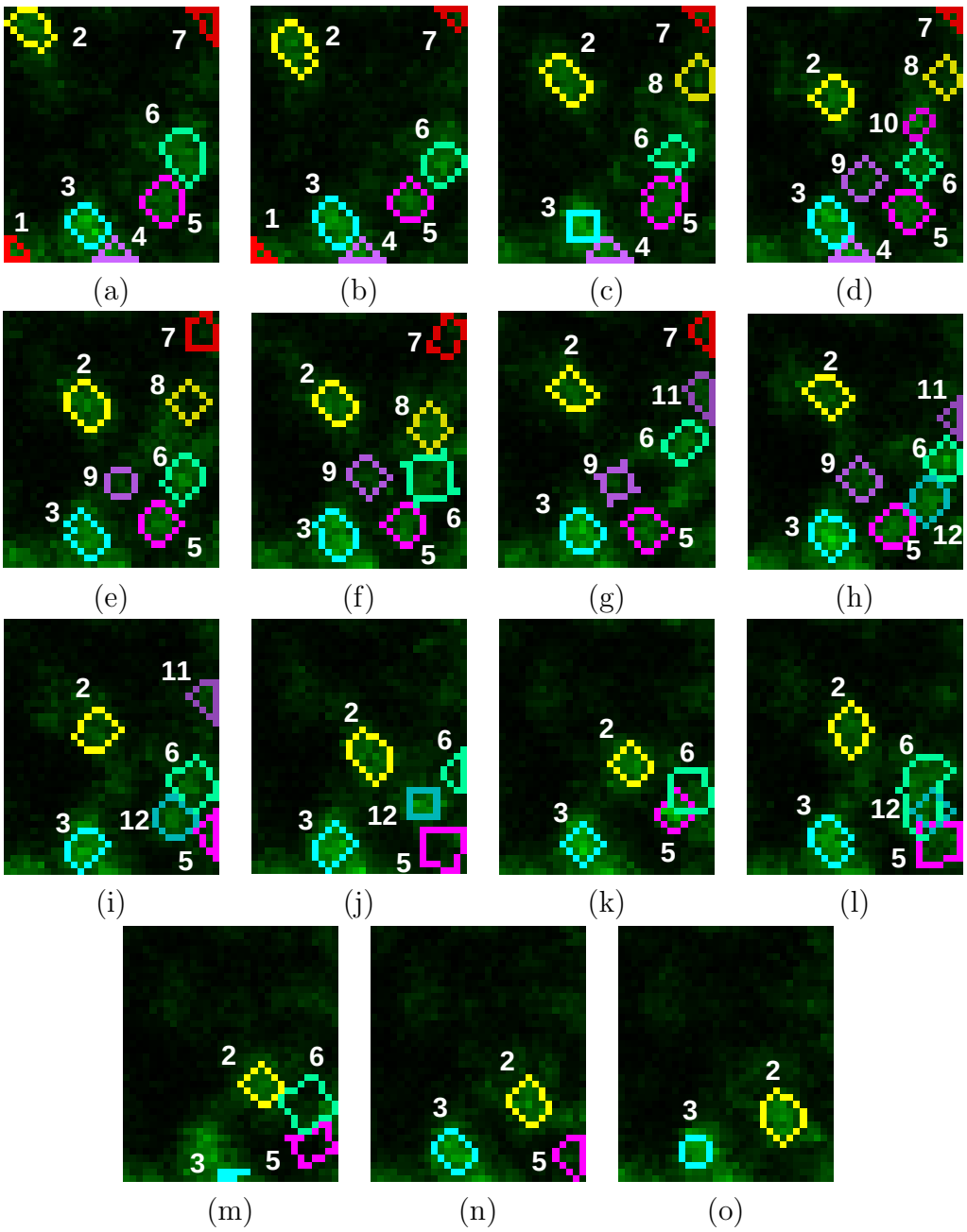


Figure 3.15. Tracked object image superimposed on original image sequence two. Different objects are delimitated by different color boundaries. (a) to (o) are image frames at time 0 sec, 1.6 sec, 3.2 sec, 4.8 sec, 6.4 sec, 8.0 sec, 9.6 sec, 11.2 sec, 12.8 sec, 14.4 sec, 16.0 sec, 17.6 sec, 19.2 sec, 20.8 sec, and 22.4 sec.

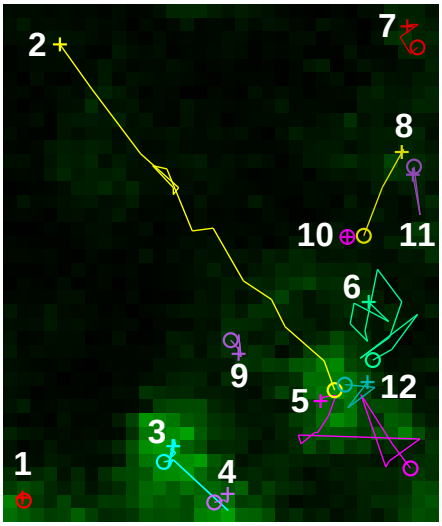


Figure 3.16. Object trajectories. The trajectories of different objects are drawn with different colors. The "+" sign and "○" sign indicate the start and end positions of each trajectory, respectively.

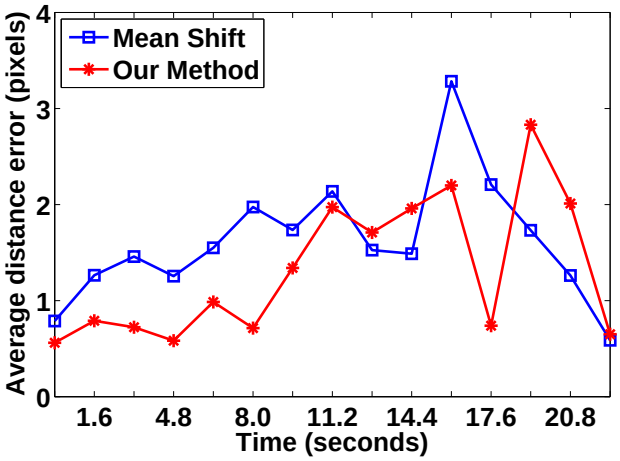


Figure 3.17. The time series plot shows the comparison of the average distance from ground truth in pixels (averaged per substructure at each frame time) between our method and mean shift multiple object tracking method.

"Missing" objects means that an object is in the ground truth and is not tracked by our method like object O1 after 3.2 second and object O4 after 6.4 second. The two missing objects are due to un-stability at the image boundary. "Incorrect" means an object is not in the ground truth and is tracked by our method like object O12 between 12.8 seconds and 14.4 seconds. Objects with long life spans, such as objects O2, O3, O5, and O6, are reliably detected.

As a comparison study, we carried out the tracking experiment by using mean shift method [65] that is recognized for good performance in non-rigid object tracking (Fig. 3.17 and Fig. 3.19). The original mean shift object tracking method only tracks one object and can not deal with a variable number of multiple objects with object appearing and disappearing. We extend it by combining our multiple object detection method to track multiple object and use distance threshold to prevent two objects from falling to the same location. As can be seen from Fig. 3.17, the overall average distance error is 1.2101 pixels for our method compared to 1.5939 pixels for mean shift multiple object tracking method. When compared on the tracked number of subcellular structures, our method is better than mean shift method. The mean shift method tends to retain objects which is reflected as the increasing number of incorrectly detected object in Fig. 3.19.

The initialization of state transition density of $(l, w, \theta, x, y)^T$ is set to be a Gaussian distribution with zero mean vector and covariance matrix $\Sigma = [0.5, 0, 0, 0, 0; 0, 0.5, 0, 0, 0; 0, 0, 1, 0, 0; 0, 0, 0, 20, 0; 0, 0, 0, 0, 20]$. The random walk covariance matrix of $(l, w, \theta, x, y)^T$ is $\Sigma = [0.1, 0, 0, 0, 0; 0, 0.1, 0, 0, 0; 0, 0, 0.1, 0, 0; 0, 0, 0, 3, 0; 0, 0, 0, 0, 3]$. For the augmented variable z, $p(z_{t.i}|z_{(t-1).i})$ is set to be a uniform distribution on $(0, 1)$. p_d, p_a, p_u, p_s, and p_h are set to be 0.1, 0.1, 0.6, 0.1, and 0.1, respectively. $\lambda_a = 7$, $\lambda_m = 5$, and $\lambda_d = 0.4$. For RJMCMC sample generation, we set sample number $N = 800$ in the experiment. P_{new} is set to be 0.005. The covariance matrix

of (l, w, θ, x, y) for the Gaussian distribution $p(\mathbf{X}_{n_t.j}|\mathbb{M}_{t.j})$ is set as $\Sigma = [1, 0, 0, 0, 0;$ $0, 1, 0, 0, 0;$ $0, 0, 0.1, 0, 0;$ $0, 0, 0, 2, 0;$ $0, 0, 0, 0, 2]$.

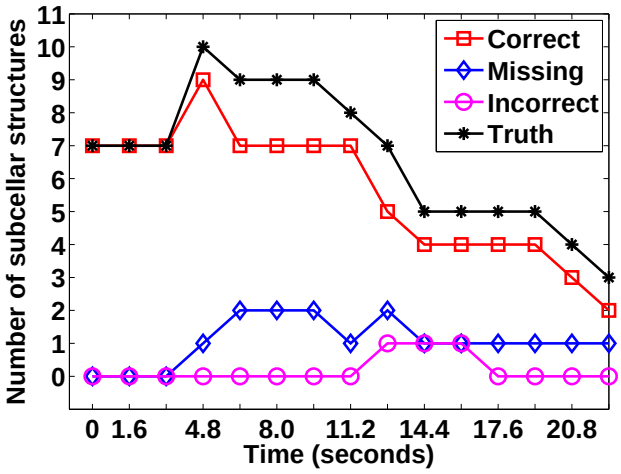


Figure 3.18. The time series of the number of tracked object by our method. The truth means that the real number of object. The correct tracked object means that object is in the ground truth and tracked by our method. The missing object means that object is in the ground truth and not tracked by our method. The incorrect means object is not in the ground truth and is tracked by our method.

3.10 Conclusions

A framework based on sequential Monte Carlo (SMC) method is presented in this chapter for automatic multiple interacting subcellular structure tracking. To model the interaction between objects, we augment objects representation in the 2D plane by an extra dimension and evaluate the overlapping relationship in the 3D space. RJMCMC algorithm with a novel height swap move is applied to sample the distribution of the varying dimension joint state efficiently. Different multiple object interaction modalities are detected and handled during the tracking process. The experiment results show that our method is feasible and effective. Although object splitting and merging are not explicitly modeled in our framework, the experiment

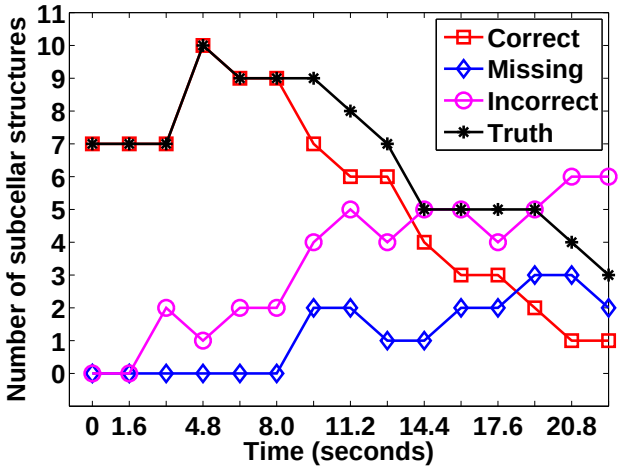


Figure 3.19. The time series of the number of tracked object by mean shift multiple object tracking method. The truth means that the real number of object. The correct tracked object means that object is in the ground truth and tracked by our method. The missing object means that object is in the ground truth and not tracked by our method. The incorrect means object is not in the ground truth and is tracked by our method.

results show that by modeling the object interaction in 3D space, our method can detect such object interacting event correctly.

CHAPTER 4

TRACKING VARIABLE NUMBER OF MULTIPLE SUBCELLULAR STRUCTURES IN 3D+T

4.1 Introduction

In Chapter 2 and Chapter 3, the image sequences we worked on are 2D plane image stacked in time order. We call it 2D+T image data. In this section, we are going to deal with 3D volume image stacked in time order. At each frame time, the LSCM provides 3D subcellular volume information about the specimen. We call this kind of image stack sequences as 3D+T volume data. A 3D subcellular volume taken by LSCM is shown in Fig. 4.1. As can be seen from Fig. 4.1, tracking subcellular structure in 3D space has the following challenges:

- Variable object size, ranging from very large to very small;
- Changing number of objects caused newly-exited or dying out of fluorophore;
- Large shape deformation due to relatively long time elapse between adjacent frames;
- Lacking of stable object features resulted from inconstant fluorescence;
- Diverse motion modalities, such as Brownian motion and translational motion;
- Cluttered environment and high noise due to photo bleaching.

The subcellular structure tracking for 3D+T volume data follows the sequential Monte Carlo (SMC) framework of the 2D+T tracking method. In this chapter, we focus on how to generate the state representation for the 3D subcellular structure.

The rest of this chapter is organized as follows. In Section 4.2, we represent the variable number of multiple 3D subcellular structures by joint state. The RJMCMC

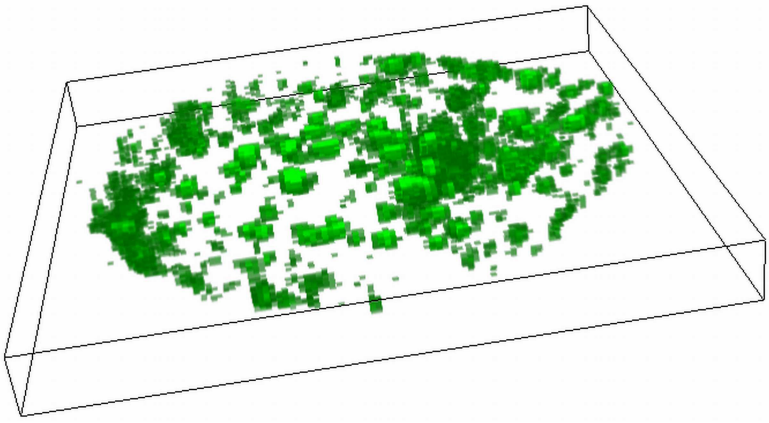


Figure 4.1. The volume rendering of a 3D green fluorescent protein (GFP) volume obtained from the laser scanning confocal microscope (LSCM). The unit ratio of x, y, and z axes is x:y:z=5:5:1.

sampling method for 3D+T tracking is detailed in Section 4.3. The detection and representation of 3D subcellular structure are described in Section 4.4. We introduce the marker residual volume guided object appearance model in Section 4.5, and give the observation mode in Section 4.6. The summary of the whole algorithm for 3D subcellular structure tracking is described in Section 4.7. Finally experimental results and conclusions are presented in Section 4.8 and Section 4.9.

4.2 Modeling Variable Number of 3D Subcellular Structures

4.2.1 Joint State Space Representation

The joint state $\mathbf{X}$ of multiple 3D subcellular structures at time t is denoted as:

$$\mathbf{X}_t = \{\mathbf{X}_{t.i} | i \in n_t\}, \tag{4.1}$$

where n_t, with its cardinality $|n_t| \geq 1$, is the set of object identity number indicating which objects contribute to represent the joint state. The object i of interest is represented as an oriented bounding volume (OBV) with state $\mathbf{X}_i = (x, y, z, l, w, h, \gamma, \beta, \alpha)^T$, where, (x, y, z) is the center of the OBV in the Cartesian coordinate system

of the image volume, (l, w, h) are the length, width, and height of the OBV, (γ, β, α) are the rotation angles about the $\hat{x}$, $\hat{y}$, and $\hat{z}$ axes which pass through the OBV center and parallel to x, y, and z axes, respectively.

Similar to the 2D+T image sequence case, we treat the background as an object in our joint state model and introduce its state at time t as $\mathbf{X}_{t.0} = (x_0, y_0, z_0, l_0, w_0, h_0, \gamma_0, \beta_0, \alpha_0)^T$, in order to include the background information more naturally into the calculation of the observation likelihood. The inclusion of the background information into the likelihood calculation can facilitate the prevention of objects from being attracted to the same high likelihood location when there are other candidate locations available and the determination of object appearing and disappearing. Since the state of the background normally does not change in the same 3D volume sequence, it can be treated as a constant vector. The background state transition probability therefore is:

$$p(\mathbf{X}_{(t+1).0}|\mathbf{X}_{t.0}) = p(\mathbf{X}_{(t+1).0} = (x_0, y_0, z_0, l_0, w_0, h_0, \gamma_0, \beta_0, \alpha_0)^T) = 1. \tag{4.2}$$

4.2.2 Joint State Transition Density

Given state parameter vector $\mathbf{X}_t$ and object identity indicator set n_t, the joint distribution is denoted as $p(n_t, \mathbf{X}_{n_t})$. The general state transition density function $p(\mathbf{X}_{n_t}|\mathbf{X}_{n_{t-1}})$ can be re-written as $p(n_t, \mathbf{X}_{n_t}|n_{t-1}, \mathbf{X}_{n_{t-1}})$ and is further factorized as following:

$$p(n_t, \mathbf{X}_{n_t}|n_{t-1}, \mathbf{X}_{n_{t-1}}) = p(\mathbf{X}_{n_t}|n_t, n_{t-1}, \mathbf{X}_{n_{t-1}})p(n_t|n_{t-1}, \mathbf{X}_{n_{t-1}}), \tag{4.3}$$

where $p(n_t|n_{t-1}, \mathbf{X}_{n_{t-1}})$ is the probability distribution of n_t conditioned on n_{t-1} and $\mathbf{X}_{n_{t-1}}$, modeling the change of the number of the objects, $p(\mathbf{X}_{n_t}|n_t, n_{t-1}, \mathbf{X}_{n_{t-1}})$ models the joint state parameters of all the objects in set n_t. Although collisions between 3D subcellular structures can happen in real, we treat such effects as the noise in the

system equation and assume the independence between the individual states of each object. Thus $p(\mathbf{X}_{n_t}|n_t, n_{t-1}, \mathbf{X}_{n_{t-1}})$ can be factorized as:

$$p(\mathbf{X}_{n_t}|n_t, n_{t-1}, \mathbf{X}_{n_{t-1}}) \triangleq \prod_{j\in\mathcal{B}_t} p(\mathbf{X}_{n_t.j}) \prod_{i\in\mathcal{S}_t} p(\mathbf{X}_{n_t.i}|\mathbf{X}_{(n_{t-1}).i}), \tag{4.4}$$

where $\mathcal{S}_t = n_t \cap n_{t-1}$ is the set of objects at time $t-1$ that remain at time t, $\mathcal{B}_t = n_t \backslash n_{t-1}$ is the set of new objects that are not in set n_{t-1}. Probability distribution $p(\mathbf{X}_{n_t.j})$ will be discussed in Section 4.3.3.

4.3 RJMCMC Method for 3D+T Tracking Sample Generation

4.3.1 Acceptance Ratio for RJMCMC in SMC Framework for 3D+T Tracking

Following the introduction of RJMCMC acceptance ratio in Chapter 3, the general form of RJMCMC acceptance ratio of SMC method for 3D+T tracking is as:

$$\alpha(n'_t,\mathbf{X}'_{n_t};n_t,\mathbf{X}_{n_t})=\frac{p(\mathbf{Z}_t|n'_t,\mathbf{X}'_{n_t})p(n'_t,\mathbf{X}'_{n_t}|\mathbf{Z}_{1:t-1})p_{m'}q_{m'}(n_t,\mathbf{X}_{n_t};n'_t,\mathbf{X}'_{n_t})}{p(\mathbf{Z}_t|n_t,\mathbf{X}_{n_t})p(n_t,\mathbf{X}_{n_t}|\mathbf{Z}_{1:t-1})p_m q_m(n'_t,\mathbf{X}'_{n_t};n_t,\mathbf{X}_{n_t})}\left|\frac{\partial(\mathbf{X}'_{n_t},\mathbf{U}')}{\partial(\mathbf{X}_{n_t},\mathbf{U})}\right|, \tag{4.5}$$

where $\mathbf{U}$ and $\mathbf{U}'$ are the auxiliary random variable vectors guarantying that the mapping from $(\mathbf{X}_{n_t}, \mathbf{U})$ to $(\mathbf{X}'_{n_t}, \mathbf{U}')$ is a one-to-one mapping, which is so called dimension matching between $\mathbf{X}_{n_t}$ and $\mathbf{X}'_{n_t}$. p_m is the move specified probability and q_m is the proposal function for $\mathbf{U}$, where m, $m \in \{u, s, d, a\}$, represents the move types with u, s, d, and a corresponding to update move, identity swap move, disappear move, and appear move, respectively. The sum of the move probabilities is set to be one as $p_u + p_s + p_d + p_a = 1$. The last term of equation (4.5) is the Jacobian of the one-to-one mapping from $(\mathbf{X}_{n_t}, \mathbf{U})$ to $(\mathbf{X}'_{n_t}, \mathbf{U}')$. In this chapter, we present how to design both the trans-dimensional and within-dimensional moves of RJMCMC in a way such that the Jacobian term is always equal to one. Reader please see Appendix A for details.

4.3.2 RJMCMC Move Proposals for 3D+T Tracking

We apply four moves for the 3D+T object tracking, namely update move, identity swap move, disappear move, and appear move. These four moves can be classified into within-dimensional move and trans-dimensional move. The within-dimensional move will not change the dimension of the joint state $\mathbf{X}_{n_t}$, while Trans-dimensional move will. Update move and identity swap moves belong to the first category, and disappear move and appear move belong to the second one. For the update move, we propose a new joint state by updating the state of only one 3D subcellular structure using random walk [60]. Identity swap move is to propose a new joint state by switching the states of two 3D subcellular structures. In the disappear move, we propose a new joint state by deleting the state of one 3D subcellular structure from the current joint state. Vice versa, the appear move proposes a new joint state by adding the state of one 3D subcellular structure to the current joint state.

4.3.2.1 Update Move

For the update move, an object identity number i is uniformly selected from the current identification number set n_t, and random walk is applied to it. The update move proposal is:

$$q_u(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = q_u(i) q_u(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}, i), \tag{4.6}$$

where $q_u(i) = 1/|n_t|$ is the proposal distribution for selecting i, $i \in \{n_t \backslash 0\}$, $q_u(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}, i) = q_u(\mathbf{U})$, where $q_u(\mathbf{U})$ is a Gaussian distribution. The deterministic mapping relation from $(\mathbf{X}_{n_t}, \mathbf{U})$ to $(\mathbf{X}'_{n_t}, \mathbf{U}')$ is as $\{\mathbf{U}' = -\mathbf{U}, \mathbf{X}'_{n_t.i} = \mathbf{X}_{n_t.i} + \mathbf{U}\}$, $i \in \{n_t \setminus 0\}$, and $\{\ldots, \mathbf{X}'_{n_t.j} = \mathbf{X}_{n_t.j}, \ldots\}$, $j \in \{n_t \setminus i\}$.

4.3.2.2 Identity Swap Move

For the identity swap move, two objects i and j in the current object set n_t are uniformly selected and their identities are exchanged. The identity swap move proposal is:

$$q_s(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = q_s(i,j) q_s(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}, i, j), \tag{4.7}$$

where $q_s(i,j) = 1/\binom{|n_t|}{2}$ is the proposal distribution for selecting the pair (i,j), and $q_s(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}, i, j) = 1$. The deterministic mapping from $\mathbf{X}_{n_t}$ to $\mathbf{X}'_{n_t}$ is as $\{\mathbf{X}'_{n_t.i} = \mathbf{X}_{n_t.j}, \mathbf{X}'_{n_t.j} = \mathbf{X}_{n_t.i}\}$, $i \in \{n_t \setminus 0\}$, $j \in \{n_t \setminus 0\}$, and $\{\ldots, \mathbf{X}'_{n_t.k} = \mathbf{X}_{n_t.k}, \ldots\}$, $k \in \{n_t \setminus i \setminus j\}$.

4.3.2.3 Disappear Move

For the disappear move, an object with identity number i is uniformly selected from the current identification number set n_t and its individual state is deleted from the joint state. The disappear move proposal is:

$$q_d(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = q_d(i) q_d(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}, i), \tag{4.8}$$

where $q_d(i) = 1/|n_t|$ is the proposal distribution for selecting i, $i \in \{n_t \setminus 0\}$, with $|\cdot|$ as the set cardinality operator, and $q_d(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}, i) = 1$. The deterministic mapping from $\mathbf{X}_{n_t}$ to $\mathbf{X}'_{n_t}$ is as $\{\ldots, \mathbf{X}'_{n_t.j} = \mathbf{X}_{n_t.j}, \ldots\}$, where $j \in \{n_t \setminus i\}$.

4.3.2.4 Appear Move

For the appear move, an object with identity number i is uniformly selected from set $\{\mathcal{A}_t \cup \bar{n}_t \setminus n_t\}$, with $\bar{n}_t = \cup_{k=1}^{N} n_{t-1}^{(k)}$, where N is the number of samples at time $t-1$ and $\mathcal{A}_t$ is the possible new object set at time t. $\mathcal{A}_t$ is constructed by image

processing techniques and will be discussed in Section 4.5. The individual state of the identification number i is added to the joint state. The appear move proposal is:

$$q_a(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = q_a(i) q_a(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}, i), \tag{4.9}$$

with $q_a(i) = 1/|\mathcal{A}_t \cup \bar{n}_t \backslash n_t|$ as the proposal distribution for selecting i, $i \in \{\mathcal{A}_t \cup \bar{n}_t \backslash n_t\}$, $q_a(n'_t, \mathbf{X}'_{n_t};\ n_t, \mathbf{X}_{n_t}, i) = q_a(\mathbf{U})$, where $q_a(\mathbf{U})$ is the proposal of generating the state for the newly added object. The formulation of $q_a(\mathbf{U})$ will be presented in Section 4.5. The deterministic mapping relation from $(\mathbf{X}_{n_t}, \mathbf{U})$ to $\mathbf{X}'_{n_t}$ is as $\mathbf{X}'_{n_t.i} = \mathbf{U}$, $i \in \{\mathcal{A}_t \cup \bar{n}_t \setminus n_t\}$ and $\{\ldots, \mathbf{X}'_{n_t.j} = \mathbf{X}_{n_t.j}, \ldots\}$, $j \in n_t$.

4.3.3 Acceptance Ratios

Plugging the proposal for each move, we get the following acceptance ratios α_u, α_s α_d, and α_a for update move, identity swap move, disappear move, and appear move, respectively:

$$\alpha_u(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = \frac{p(\mathbf{Z}_t|n'_t, \mathbf{X}'_{n_t}) p(n'_t, \mathbf{X}'_{n_t}|\mathbf{Z}_{1:t-1})}{p(\mathbf{Z}_t|n_t, \mathbf{X}_{n_t}) p(n_t, \mathbf{X}_{n_t}|\mathbf{Z}_{1:t-1})}, \tag{4.10}$$

$$\alpha_s(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = \frac{p(\mathbf{Z}_t|n'_t, \mathbf{X}'_{n_t}) p(n'_t, \mathbf{X}'_{n_t}|\mathbf{Z}_{1:t-1})}{p(\mathbf{Z}_t|n_t, \mathbf{X}_{n_t}) p(n_t, \mathbf{X}_{n_t}|\mathbf{Z}_{1:t-1})}, \tag{4.11}$$

$$\alpha_d(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = \frac{p(\mathbf{Z}_t|n'_t, \mathbf{X}'_{n_t}) p(n'_t, \mathbf{X}'_{n_t}|\mathbf{Z}_{1:t-1}) p_a q_a(i) q_a(\mathbf{U}')}{p(\mathbf{Z}_t|n_t, \mathbf{X}_{n_t}) p(n_t, \mathbf{X}_{n_t}|\mathbf{Z}_{1:t-1}) p_d q_d(i)}, \tag{4.12}$$

$$\alpha_a(n'_t, \mathbf{X}'_{n_t}; n_t, \mathbf{X}_{n_t}) = \frac{p(\mathbf{Z}_t|n'_t, \mathbf{X}'_{n_t}) p(n'_t, \mathbf{X}'_{n_t}|\mathbf{Z}_{1:t-1}) p_d q_d(i)}{p(\mathbf{Z}_t|n_t, \mathbf{X}_{n_t}) p(n_t, \mathbf{X}_{n_t}|\mathbf{Z}_{1:t-1}) p_a q_a(i) q_a(\mathbf{U})}. \tag{4.13}$$

For the evaluation of $p(n_t, \mathbf{X}_{n_t}|\mathbf{Z}_{1:t-1})$, we use its mixture approximation as in equation (3.7):

$$\hat{p}(n_t, \mathbf{X}_{n_t}|\mathbf{Z}_{1:t-1}) = \frac{1}{N} \sum_{k=1}^{N} p(n_t, \mathbf{X}_{n_t}|n_{t-1}^{(k)}, \mathbf{X}_{n_{t-1}}^{(k)}), \tag{4.14}$$

where $p(n_t, \mathbf{X}_{n_t}|n_{t-1}^{(k)}, \mathbf{X}_{n_{t-1}}^{(k)})$ is evaluated by using equation (3.11) and equation (3.12) as:

$$p(n_t, \mathbf{X}_{n_t}|n_{t-1}^{(k)}, \mathbf{X}_{n_{t-1}}^{(k)}) = p(n_t|n_{t-1}^{(k)}, \mathbf{X}_{n_{t-1}}^{(k)}) \prod_{j \in \mathcal{B}_t} p(\mathbf{X}_{n_t.j}) \prod_{i \in \mathcal{S}_t} p(\mathbf{X}_{n_t.i}|\mathbf{X}_{(n_{t-1}).i}^{(k)}), \tag{4.15}$$

with $\mathcal{B}_t = n_t \setminus n_{t-1}^{(k)}$, $\mathcal{S}_t = n_t \cap n_{t-1}^{(k)}$ as previously introduced in section 3.3. There are two cases for object identification number j in $\mathcal{B}_t$, $j \in \mathcal{A}_t$ and $j \notin \mathcal{A}_t$. $\mathcal{A}_t$ is the possible new object set at time t as mentioned in appear move. For the first case, $p(\mathbf{X}_{n_t.j}) = p_{new}$, with $j \in \mathcal{A}_t$. For the second case, $p(\mathbf{X}_{n_t.j})$ is defined to be $p(\mathbf{X}_{n_t.j}) \triangleq \sum_{i \in N_{t-1}^{(j)}} p(\mathbf{X}_{n_t.j}|\mathbf{X}_{n_{t-1}.j}^{(i)})/|N_{t-1}^{(j)}|$, with $N_{t-1}^{(j)} = \{i : j \in n_{t-1}^{(i)}\}$, the set of samples at time $t-1$ containing object j. $n_{t-1}^{(i)}$ is the object set for particle i at time $t-1$. $p(n_t|n_{t-1}^{(k)}, \mathbf{X}_{n_{t-1}}^{(k)})$ is defined to be equivalent for all n_t, $n_{t-1}^{(k)}$, and $\mathbf{X}_{n_{t-1}}^{(k)}$, since we assume the equal chance of each object set n_t.

4.4 Automatic 3D Subcellular Structure Detection and Representation

4.4.1 Automatic 3D Subcellular Structure Detection

We use the regional maxima to label and detect 3D subcellular structure. For an volume at time t with intensity as I_{3t} and a 3D structuring element as S_3, the 3D gray-scale reconstructive opening operation $\circledcirc_3$ is represented as:

$$I_{3t} \circledcirc_{C_3} S_3 = I_{3t} \odot_{C_3} (I_{3t} \circ_3 S_3), \tag{4.16}$$

where $\odot_3$ is the 3D gray scale reconstruction operation, $\circ_3$ is the 3D gray scale morphological opening, and C_3 is the 3D connectivity definition for $\odot_3$. We use the regional maxima of $I_{3t} \circledcirc_{C_3} S_3$ to identify each object. Since the size of object varies a lot, from medium (about 100 voxels) to small (about 10 voxels), we apply the 6-connected connectivity as the 3D structure element, which is iso-tropic in x, y and z axis directions. Each of the regional maxima of $I_{3.t} \circledcirc_{C_3} S_3$ is dilated once to include

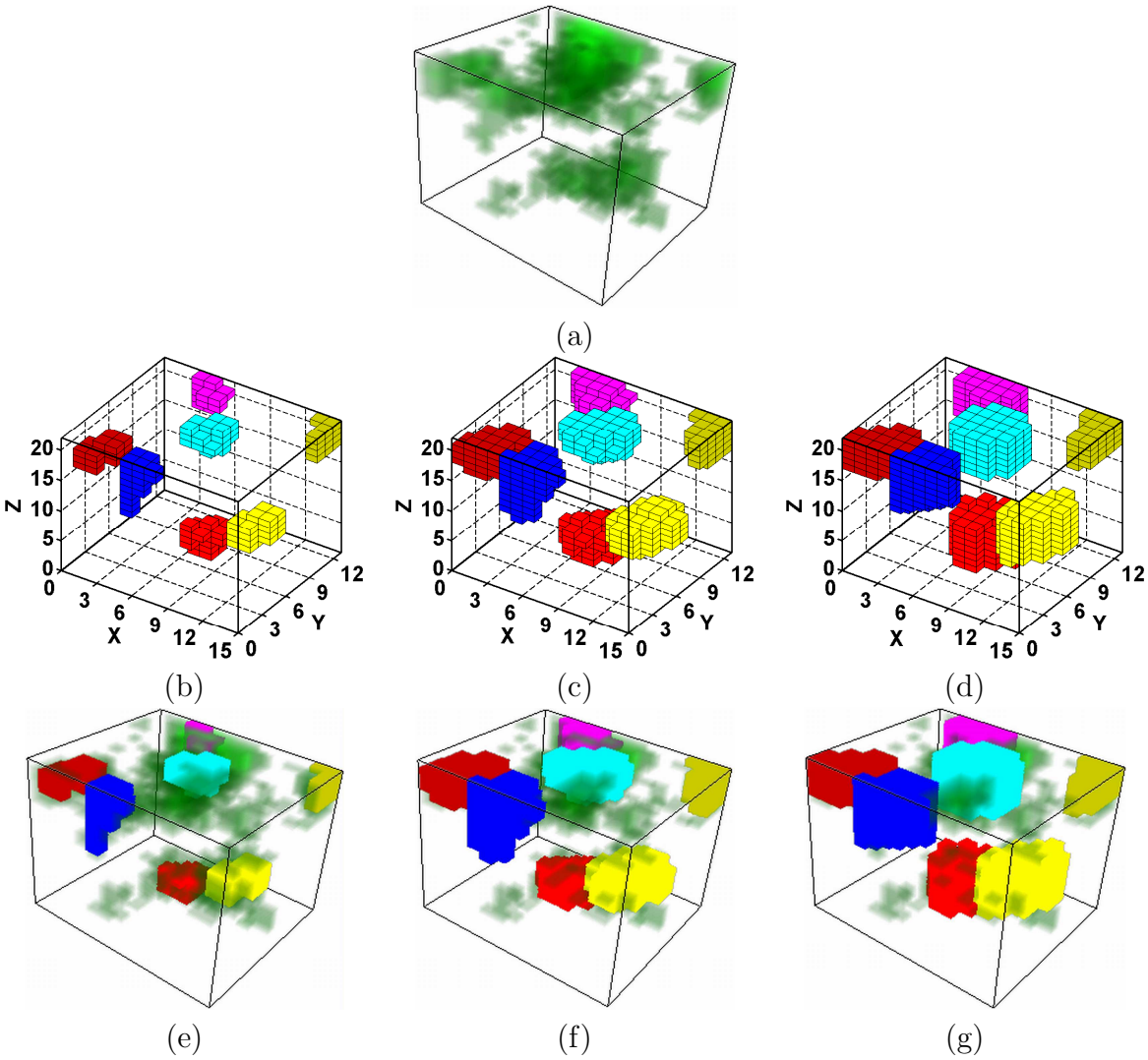


Figure 4.2. From regional maxima to state. (a) is the original volume. (b) is the regional maxima volume. (c) is the marker volume. (d) is the state volume. (e), (f), and (g) are regional maxima volume, marker volume, and state volume embedded in the original volume. The unit ratio of x, y, and z directions is x:y:z=5:5:1.

more voxel information. The corresponding binary volume is called marker volume $M_{3.t}$.

An example of regional maxima and marker volume are shown in Fig. 4.2(b) and Fig. 4.2(c), with Fig. 4.2(e) and Fig. 4.2(f) as the results of embedding them into the original volume, respectively.

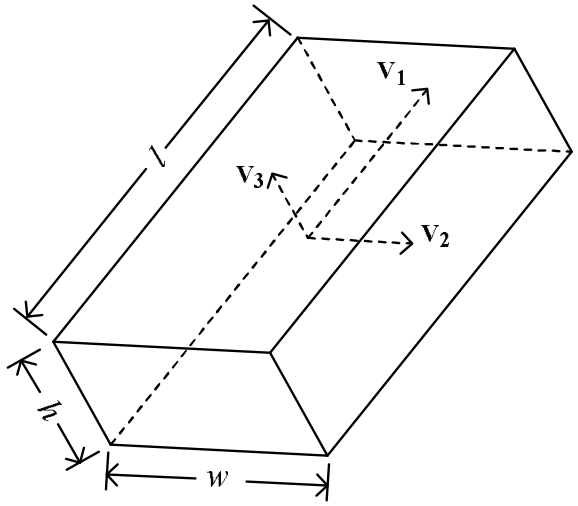


Figure 4.3.Illustration of the oriented bounding volume.

4.4.2 3D Subcellular Structure Representation

Suppose a single voxel in one subcellular object is represented as a 3D vector $\boldsymbol{u} = (x, y, z)^T$, where x, y, z are the coordinates in the 3D space and T is the transpose operation, the center position vector $\bar{\boldsymbol{u}} = [\bar{x}, \bar{y}, \bar{z}]^T$ of the region can be calculated as:

$$\bar{\boldsymbol{u}} = \frac{1}{L} \sum_{i=1}^{L} \boldsymbol{u}_i, \tag{4.17}$$

where L is the number of voxel in the marker volume. The covariance matrix of the position vector $\boldsymbol{u}$ for the marker volume can be obtained by:

$$\Sigma_p = \frac{1}{L} \sum_{i=1}^{L} \boldsymbol{u}_i \boldsymbol{u}_i^T - \bar{\boldsymbol{u}} \bar{\boldsymbol{u}}^T. \tag{4.18}$$

The eigenvalue of covariance matrix Σ_p are λ_1, λ_2, and λ_3, with their corresponding eigenvectors v_1, v_2, and v_3, respectively. Without lost of generality, we let $\lambda_1 \geq \lambda_2 \geq \lambda_3$. The eigenvectors v_1, v_2, and v_3 determine the OBV orientation of the subcellular structure. The length l, width w, and height h of OBV are determined by the distance of the projection of the object to the corresponding eigenvectors v_1, v_2,

and v_3, respectively. As illustrated in Fig. 4.3, the shape and orientation OBV can be described fully by v_1, v_2, v_3, l, w, and h.

An example of the state volume generated from marker volume is illustrated in Fig. 4.2(d), It is embedded into the original volume as Fig. 4.2(g). As can be seem from Fig. 4.2(d), the state volume can keep the 3D subcellular structure labeled correctly with the benefit of keeping the description of the 3D structure mathematically simple.

The rotation matrix produced by the eigenvectors v_1, v_2, and v_3 needs to be converted to rotation angle γ, β, and α which are used in the state representation. Let $v_1 = [v_{1.x}, v_{1.y}, v_{1.z}]^T$, $v_2 = [v_{2.x}, v_{2.y}, v_{2.z}]^T$, and $v_3 = [v_{3.x}, v_{3.y}, v_{3.z}]^T$, and suppose that the major, medium, minor axes of the 3D object have the same directions as x, y, and z axes originally, to let the object has the new orientation described by v_1, v_2 and v_3, the angles that the object rotates about the $\hat{x}$, $\hat{y}$, and $\hat{z}$ axes, which pass through the center of OBV and parallel to x, y, and z axes are γ, β, α, respectively. Here we suppose that the OBV center is fixed. We use the following formula for the conversion between the eigenvectors and rotation angles [66]:

$$\begin{bmatrix} \cos\alpha\cos\beta & \cos\alpha\sin\beta\sin\gamma - \sin\alpha\cos\gamma & \cos\alpha\sin\beta\cos\gamma + \sin\alpha\sin\gamma \\ \sin\alpha\cos\beta & \sin\alpha\sin\beta\sin\gamma + \cos\alpha\cos\gamma & \sin\alpha\sin\beta\cos\gamma - \cos\alpha\sin\gamma \\ -\sin\beta & \cos\beta\sin\gamma & \cos\beta\cos\gamma \end{bmatrix} = \begin{bmatrix} r_{1.x} & r_{2.x} & r_{3.x} \\ r_{1.y} & r_{2.y} & r_{3.y} \\ r_{1.z} & r_{2.z} & r_{3.z} \end{bmatrix}, \quad (4.19)$$

from Eq. 4.19, we get

$$\beta = \tan_2^{-1}\left(-r_{1.z}, \sqrt{r_{1.x}^2 + r_{1.y}^2}\right), \quad (4.20)$$

$$\alpha = \tan_2^{-1}\left(\frac{r_{1.y}}{\cos\beta}, \frac{r_{1.x}}{\cos\beta}\right), \quad (4.21)$$

$$\gamma = \tan_2^{-1}\left(\frac{r_{2.z}}{\cos\beta}, \frac{r_{3.z}}{\cos\beta}\right), \quad (4.22)$$

where $\tan_2^{-1}(y, x)$ is a function returning an angle by computing ordinary $\tan^{-1}(\frac{y}{x})$ and using the x and y signs to determine the quadrant of the resulting angle. Here

we only let $-\frac{\pi}{2} \leq \beta \leq -\frac{\pi}{2}$. In the case $\beta = \pm\frac{\pi}{2}$, only $\alpha \pm \gamma$, the sum or the difference of α and γ, can be computed, we set $\alpha = 0.0$ according to convention, and get

$$\beta = \frac{\pi}{2}, \tag{4.23}$$

$$\alpha = 0.0, \tag{4.24}$$

$$\gamma = \tan_2^{-1}(r_{2.x}, r_{2.y}), \tag{4.25}$$

or

$$\beta = -\frac{\pi}{2}, \tag{4.26}$$

$$\alpha = 0.0, \tag{4.27}$$

$$\gamma = -\tan_2^{-1}(r_{2.x}, r_{2.y}). \tag{4.28}$$

Reader please see Appendix C for details.

4.5 3D Marker Residual Volume-Guided Appearance Model

Same to the situation facing 2D+T tracking that the object will appear or disappear in any part of the 3D volume due to the newly-exited or dying out of fluorophore, we introduce the marker residual image-guided appearance model to solve the problem. For a 3D volume $I_{3.t}$ at time t, its marker volume $M_{3.t}$ can be denoted as:

$$M_{3.t} = \cup_{i=1}^{n} M_{3.t.i}. \tag{4.29}$$

where $M_{3t.i}$ is a binary 3D volume with segmented object i. A new image called marker residual volume $\mathbb{M}_{3.t}$ that depicts the newly appearing objects is defined as:

$$\mathbb{M}_{3.t} = \{M_{3.t.i} : M_{3.t.i} \cap M_{3.(t-1)} = \emptyset, i = 1, \cdots, n\}. \tag{4.30}$$

An illustration of marker volume and marker residual volume is shown in Fig. 4.4. Marker volume $M_{3.8}$ and $M_{3.9}$ are shown as Fig. 4.4(a) and (b), respectively. Residual

volume $\mathbb{M}_{3.9}$ is shown as Fig. 4.4(c) with it embedded in the original volume as Fig. 4.4(d).

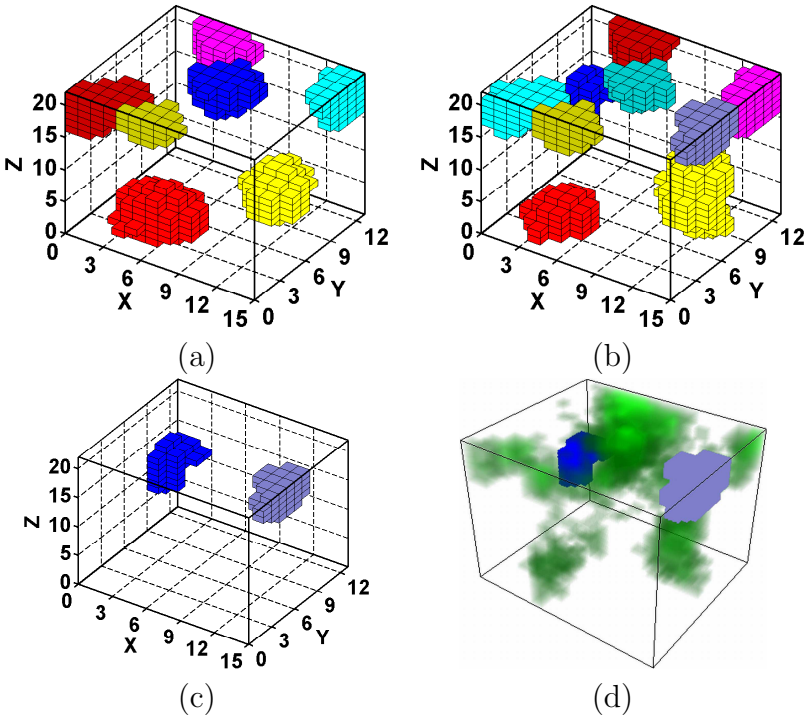


Figure 4.4. Marker residual volume generation. (a) is marker volume $M_{3.8}$. (b) is marker volume $M_{3.9}$. (c) is marker residual volume $\mathbb{M}_{3.9}$. (d) is original volume embedded with $\mathbb{M}_{3.9}$. The unit ratio of x, y, and z directions is x:y:z=5:5:1.

Each $\mathbb{M}_{3.t.i}$, the ith component in $\mathbb{M}_{3t}$, is treated as possible new object and used to form the possible appearing object set $\mathcal{A}_t$, which is mentioned in Section 4.3.2.4. The previously introduced proposal function $q_a(\mathbf{U})$ for the new object is defined as follows:

$$q_a(\mathbf{U}) = \begin{cases} p(\mathbf{X}_{n_t.j}|\mathbb{M}_{3.t.j}), & \text{if } j \in \mathcal{A}_t \\ \sum_{i \in N_{t-1}^{(j)}} P(\mathbf{X}_{n_t.j}|\mathbf{X}_{(n_{t-1}).j}^{(i)})/|N_{t-1}^{(j)}|, & \text{if } j \notin \mathcal{A}_t, \end{cases} \tag{4.31}$$

where $p(\mathbf{X}_{n_t.j}|\mathbb{M}_{3.t.j})$ generates the state $(x, y, z, l, w, h, \gamma, \beta, \alpha)$ by a Gaussian distribution centered around the object in the image $\mathbb{M}_{3.t.j}$. $N_{t-1}^{(j)} = \{i : j \in n_{t-1}^{(i)}\}$ is the set of samples at time $t-1$ containing object j.

4.6 3D Observation Model

The joint observation model is defined as:

$$p(\mathbf{Z}_t|\mathbf{X}_t) = \prod_{i \in n_t} p(\mathbf{Z}_{t.i}|\mathbf{X}_{t.i}). \tag{4.32}$$

For the measurement of each individual object state $\mathbf{X}_{t.i}$, we use the amount and intensity profile of the voxels inside its bounding volume to match with its reference $R_{t.i}$, which is obtained at the beginning of each object trajectory. Three properties between $\mathbf{X}_{t.i}$ and $R_{t.i}$ are considered in our observation measurement: (1) volume; (2) intensity mean; (3) sorted sum-of-absolute-differences (SSAD). The intuition behind the first one is that although the shape of the subcellular structure is unstable, its volume is relatively stable. The intensity mean and SSAD are used to characterize the intensity profile of voxels inside the state bounding volume, with intensity mean for the global similarity and SSAD for the similarity in detail. Thus we define the observation model for each individual object as:

$$p(\mathbf{Z}_{t.i}|\mathbf{X}_{t.i}) = p(\mathbf{Z}_{t.i}^{v}|\mathbf{X}_{t.i})p(\mathbf{Z}_{t.i}^{m}|\mathbf{X}_{t.i})p(\mathbf{Z}_{t.i}^{d}|\mathbf{X}_{t.i}), \tag{4.33}$$

with $p(\mathbf{Z}_{t.i}^{v}|\mathbf{X}_{t.i})$, $p(\mathbf{Z}_{t.i}^{m}|\mathbf{X}_{t.i})$, and $p(\mathbf{Z}_{t.i}^{d}|\mathbf{X}_{t.i})$, for volume measurement, intensity mean, and SSAD, respectively.

Suppose the set of intensity of voxels inside the bounding volume of $\mathbf{X}_{t.i}$ is $A_n = \{a_1, \ldots, a_j, \ldots, a_n\}$, where n is the amount of voxels and a_j stands for the intensity of the individual voxel. The mean intensity of A_n is:

$$I_{A_n} = \sum_{j=1}^{n} a_j/n. \tag{4.34}$$

Same to $R_{t.i}$, we have its intensity set $B_m = \{b_1, \ldots, b_j, \ldots, b_m\}$ and intensity mean $I_{B_m} = \sum_{j=1}^{m} b_j/m$, with m as the amount of voxels.

We define $p(\mathbf{Z}_{t.i}^v|\mathbf{X}_{t.i})$ as:

$$p(\mathbf{Z}_{t.i}^v|\mathbf{X}_{t.i}) \propto \exp\left(-\lambda_v \left(1 - \frac{\min(n,m)}{\max(n,m)}\right)^2\right). \tag{4.35}$$

Similarly, $p(\mathbf{Z}_{t.i}^m|\mathbf{X}_{t.i})$ is defined by:

$$p(\mathbf{Z}_{t.i}^m|\mathbf{X}_{t.i}) \propto \exp\left(-\lambda_m \left(1 - \frac{\min(I_{A_n}, I_{B_m})}{\max(I_{A_n}, I_{B_m})}\right)^2\right). \tag{4.36}$$

Here, λ_v, λ_m are hyperparameters.

The SSAD between two volumes is the same as SSAD between two regions which is the intensity sets matching problem. Finally, $p(\mathbf{Z}_{t.i}^d|\mathbf{X}_{t.i})$ is as:

$$p(\mathbf{Z}_{t.i}^d|\mathbf{X}_{t.i}) \propto \exp\left(-\lambda_d \cdot (SSAD(A_n, B_m))^2\right), \tag{4.37}$$

with λ_d as a hyperparameter too. The intuition of SSAD is that for the small subcellular structure, its salient points are the voxels with high intensity value. By comparing the intensity values of these salient points, we can get the information about how similar two small structures are, since there is no obvious spatial pattern of the voxels within such small volumes.

4.7 Summary of the Algorithm for Multiple 3D Subcellular Structure Tracking

The algorithm for 3D+T subcellular structure tracking is as follows:

1. Automatically detect the subcellular structures using marker residual volume. Estimate the state $\mathbf{X}_{0.i}$ of each individual subcellular structure, and combine them into $\mathbf{X}_0$. Sample the joint state $\mathbf{X}_0$ with N samples. Thus the initial distribution of $\mathbf{X}_0$ is approximated by $p(\mathbf{X}_0) \approx \{\mathbf{X}_0^{(s)}\}$, $s = 1, \ldots, N$. Set $t = 0$.
2. Set $t = t+1$. Draw a sample of $\mathbf{X}_t$ from the prediction density $\sum_{s=1}^{N} p(\mathbf{X}_t|\mathbf{X}_{t-1}^{(s)})/N$.

3. Apply RJMCMC method to draw samples.
   (a) Draw a sample w from the uniform distribution on $(0, 1)$.
   (b) if $0 \leq w < p_u$, apply update move. Update the state of one subcellular structure in the joint state using equation (4.6).
   (c) If $p_u \leq w < p_u + p_s$, apply identity swap move. Swap the states of two subcellular structures in the joint state using equation (4.7).
   (d) If $p_u + p_s \leq w < p_u + p_s + p_d$, apply disappear move. Delete the state of one subcellular structure from the joint state using equation (4.8).
   (e) If $p_u + p_s + p_d \leq w$, apply appear move. Add the state of one subcellular structure to the joint state using equation (4.9).
   (f) Calculate the acceptance ratio $\alpha(\mathbf{X}_t, \mathbf{X}'_t)$ based on the move type using equations (4.10), (4.11), (4.12), and (4.13).
   (g) Draw a sample w' from the uniform distribution on $(0, 1)$. If $w' < \alpha(\mathbf{X}_t, \mathbf{X}'_t)$, use $\mathbf{X}'_t$ as the new joint state; else use $\mathbf{X}_t$. Add the new state to the final sample set.
4. Go to step 2.

4.8 Experimental Results

The real 3D+T volume sequence data we used for experiment has total of 147 3D volumes, each with a size of $160 \times 140 \times 22$ voxels taken with a 0.88 seconds interval. We use a $15 \times 13 \times 22$ portion of the 3D volume spanning 7.04 seconds to evaluate our algorithm, as shown in Fig. 4.5. Here the unit ratio of x, y, and z directions is set to be 5:5:1 for better 3D rendering effect instead of the original unit ratio 10:10:1. As can be observed from Fig. 4.5, which is the volume rendering of the original volume, the subcellular structures are very close to each other with different

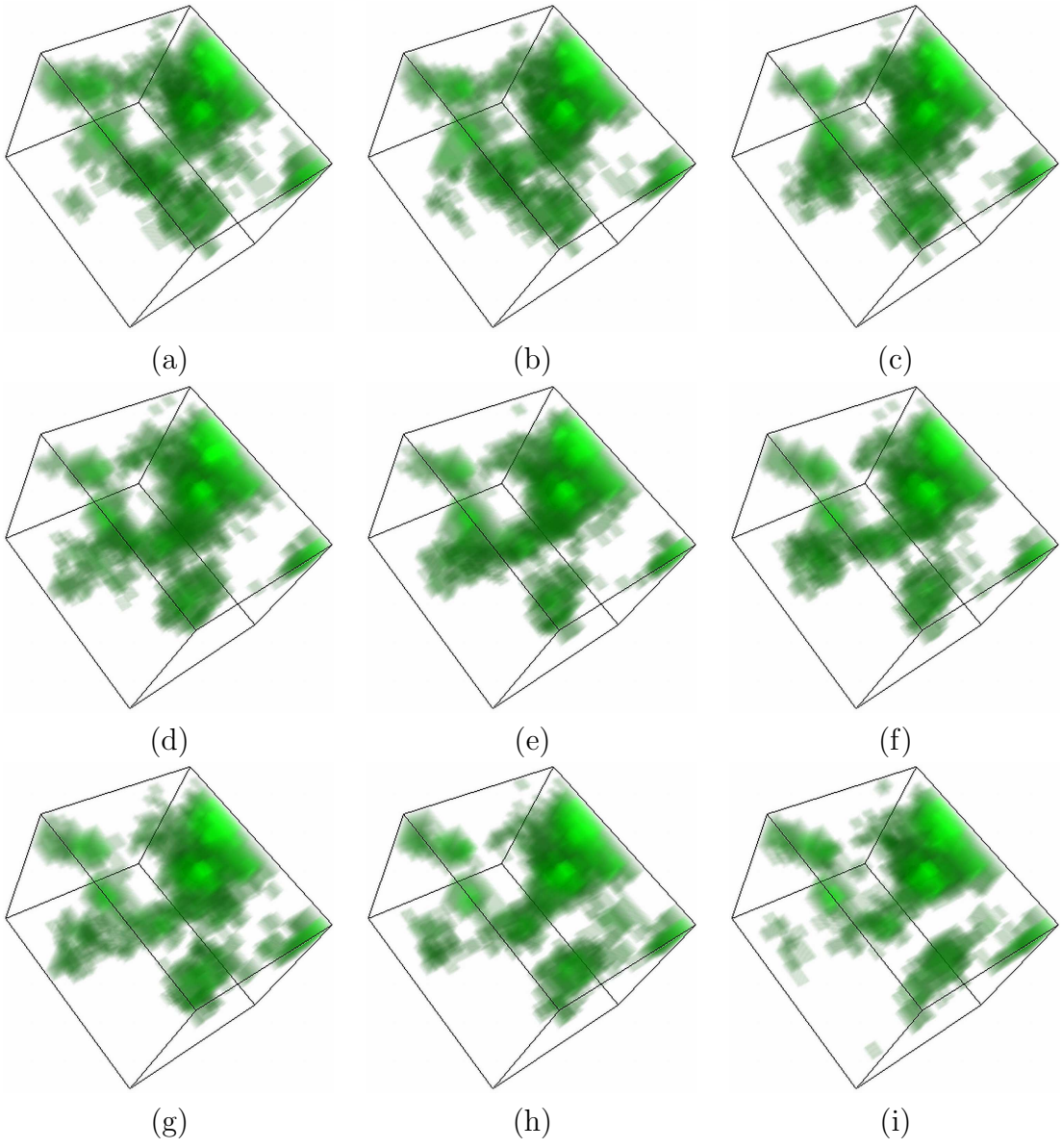


Figure 4.5. Volume rendering of a portion of the 3D green fluorescent protein (GFP) volume sequence obtained from the laser scanning confocal microscope (LSCM). (a) to (i) are volumes at time 0 sec, 0.88 sec, 1.76 sec, 2.64 sec, 3.52 sec, 4.40 sec, 5.28 sec, 6.16 sec, and 7.04 sec. The unit ratio of x, y, and z directions is x:y:z=5:5:1.

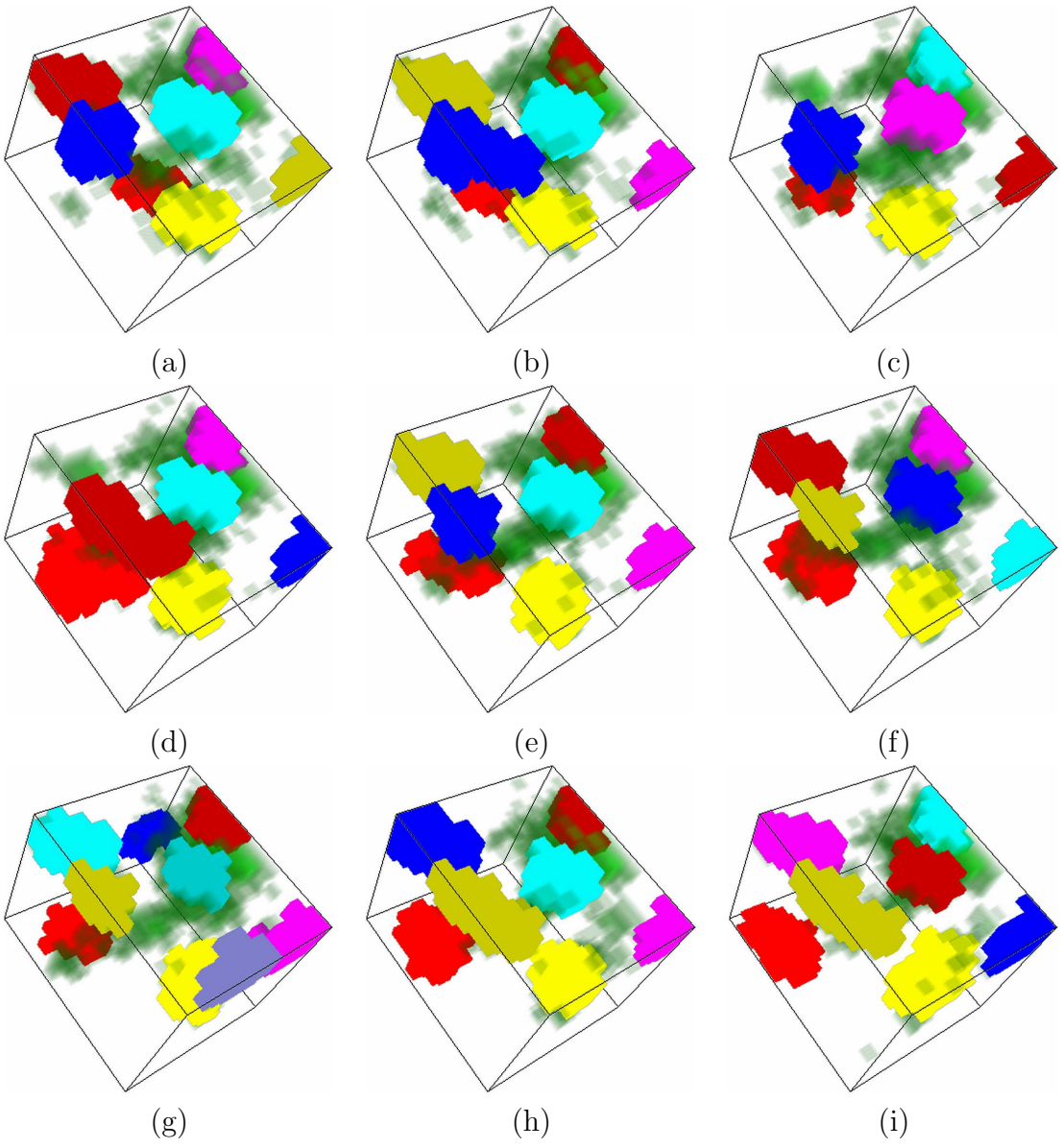


Figure 4.6. Results of marker volume embedded in the real volume sequence, with different marker volume indicated by different color. (a) to (i) are volumes at time 0 sec, 0.88 sec, 1.76 sec, 2.64 sec, 3.52 sec, 4.40 sec, 5.28 sec, 6.16 sec, and 7.04 sec. The unit ratio of x, y, and z directions is x:y:z=5:5:1.

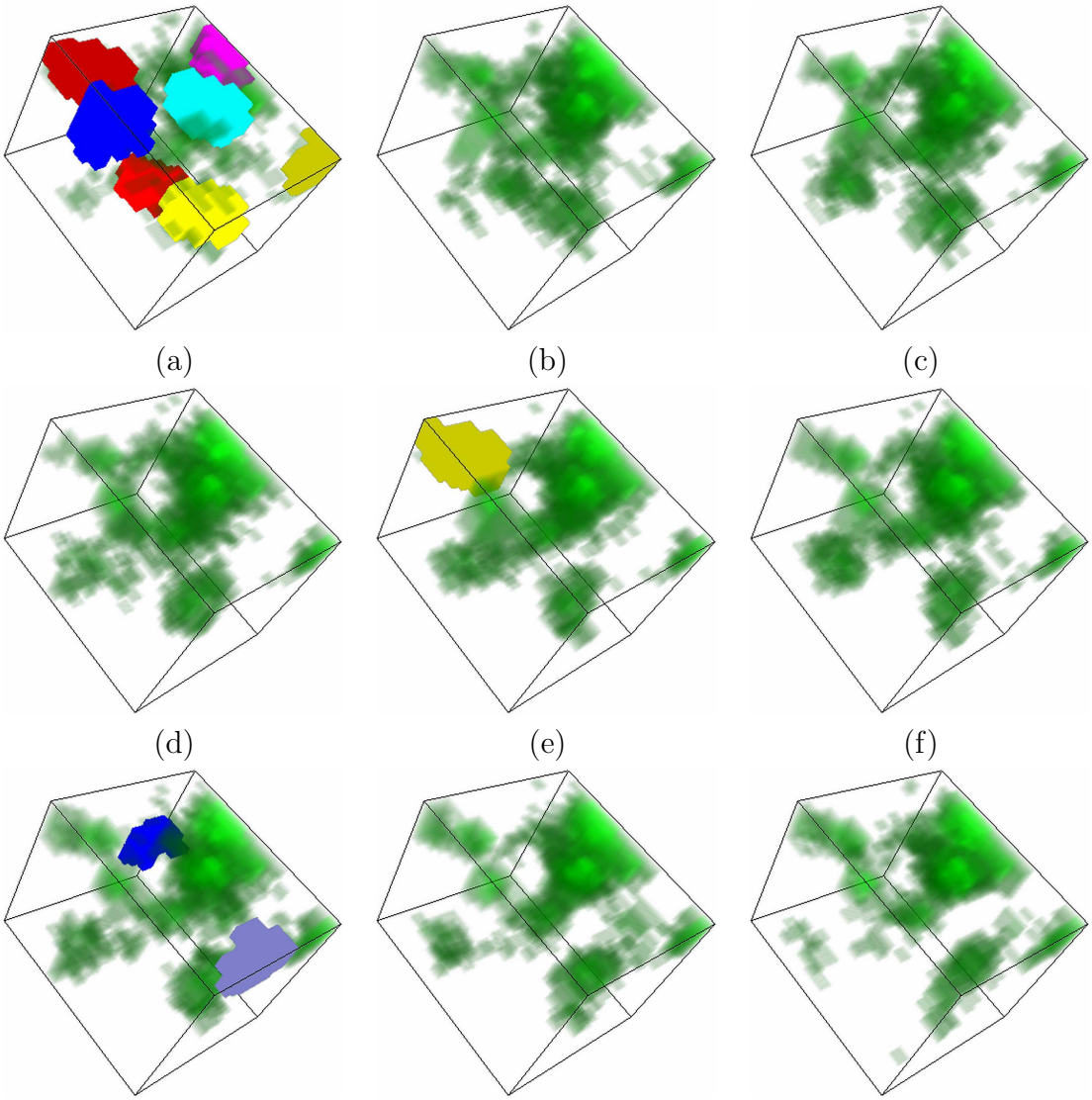


Figure 4.7. Results of marker residual volume embedded in the real volume sequence, with different marker residual volume indicated by different color. (a) to (i) are volumes at time 0 sec, 0.88 sec, 1.76 sec, 2.64 sec, 3.52 sec, 4.40 sec, 5.28 sec, 6.16 sec, and 7.04 sec. The unit ratio of x, y, and z directions is x:y:z=5:5:1.

3D shape and size. For the volume rendering, we developed a 3D animation graphical user interface (GUI) application by using Visualization ToolKit (VTK) 5.0.

4.8.1 Results of Marker Volume and Marker Residual Volume

We use expectation-maximization (EM) algorithm [64] to separate the background and foreground. The marker volumes at each volume time embedded in the original 3D volume are illustrated in Fig. 4.6. As shown in Fig. 4.6, the markers can label the 3D subcellular structures correctly. Some green parts seem not being labeled. This is because they are far away from their regional maxima and we only use regional maxima with their neighboring voxels included by the 3D dilation operation to represent the 3D subcellular structures. The marker residual volumes are also shown in Fig. 4.7, with 10 new 3D subcellular structures detected in which a total of seven, one, and two new 3D subcellular structures are detected at time 0 second, 3.52 seconds, and 5.28 seconds, respectively.

4.8.2 Tracking Results

The final tracking results of our method are shown in Fig. 4.8. Our algorithm can track different subcellular structure movement modalities such as Brownian motion and translational motion. The object appearing and disappearing phenomena are also correctly detected. Although there are 10 new 3D subcellular structures detected by marker residual volume, only nine of them are tracked by our RJMCMC method. Among the nice objects, object O1 is doing translational motion, the others are having Brownian movement. The movement of each subcellular structure is clearly demonstrated by its trajectories shown in Fig. 4.9 with different view angles. As can be seen in Fig. 4.9(b) and (d), the 3D+T volume sequence tracking results can provide more subcellular structure information than 2D+T image sequence which is

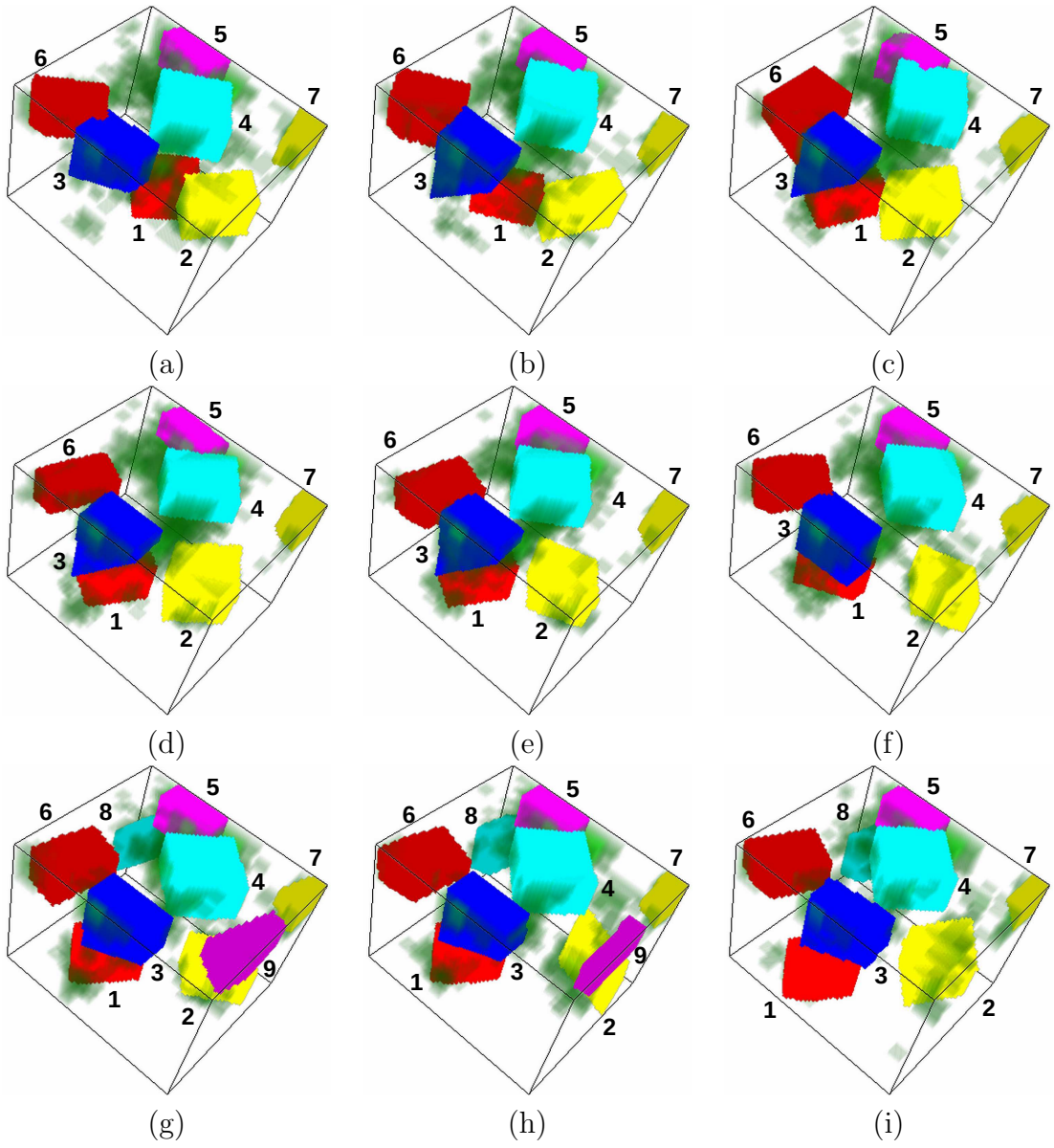


Figure 4.8. Tracking results of subcellular structures embedded in the real volume sequence, with different subcellular structure indicated by different color. (a) to (i) are volumes at time 0 sec, 0.88 sec, 1.76 sec, 2.64 sec, 3.52 sec, 4.40 sec, 5.28 sec, 6.16 sec, and 7.04 sec. The unit ratio of x, y, and z direction is x:y:z=5:5:1.

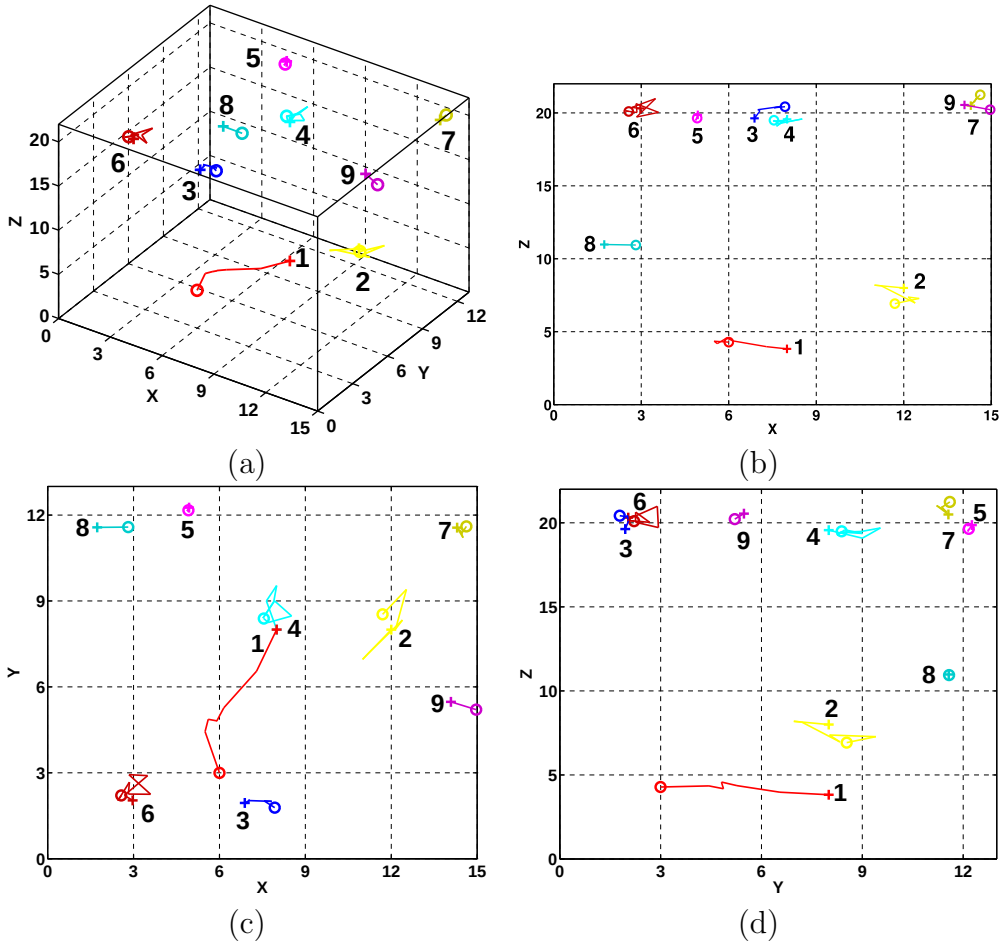


Figure 4.9. Different views of the tracked trajectory. (a) is viewed with *azimuth* = 34°, *elevation* = 32°. (b) is viewed from y axis direction. (c) is viewed from z axis direction. (d) is viewed from x axis direction. The unit ratio of x, y, and z axes is x:y:z=5:5:1.

only able to provide subcellular structure information at a fixed z level. If the acquiring z axis position is fixed at 20, the movement information of objects O1, O2, and O8 will be lost.

It worth noticing that at time 3.52 seconds the marker residual volume detects a new subcellular structure (Fig. 4.7(e)), our RJMCMC moves correctly reject this fake new object and keep object O6 alive. Two real new objects O8 and O9 detected

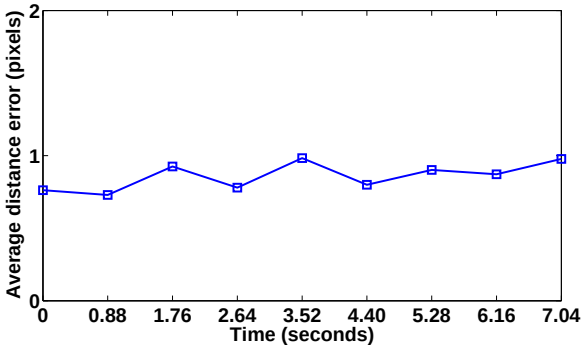


Figure 4.10.Average distance error at each time by our method.

at time 5.28 (Fig. 4.7(g)). Object O8 lasts till the end of the sequence. Object O9 just exists for 1.76 seconds at time 5.28 seconds and 6.16 seconds.

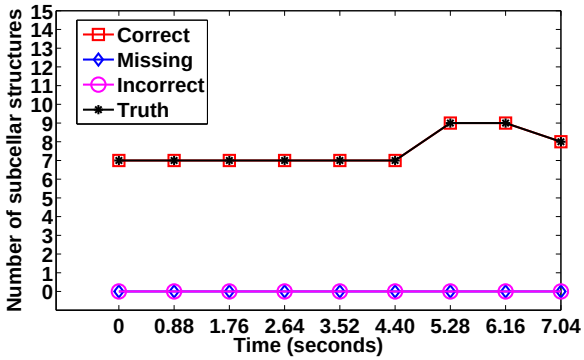


Figure 4.11.Evaluation of the number of tracked object at each time by ourmethod.

The quantitative performance of our algorithm is evaluated by the ground truth which is obtained by a human-in-the loop segmentation editor. The average displacement error between the center of the tracked subcellular structure and the center of the ground truth was calculated for each 3D volume at 0.88 seconds recording interval (Fig. 4.10). The average distance error is 0.9361 pixels for the testing sequence. Here the distance unit is in terms of the pixel in the x-y plane. Fig. 4.11 shows the change of number of objects during tracking process. "Truth" means the real number of

objects in the ground truth. "Correct" tracked objects means that an object is in the ground truth and is tracked by our method. "Missing" objects means that an object is in the ground is not tracked by our method. "Incorrect" means an object is not in the ground truth and is tracked by our method. Our method tracked the number of object during the whole volume sequence time correctly.

As for the parameters used in our method, the state transition density of $(x, y, z, l,\ w, h, \gamma, \beta, \alpha)^T$ is set to be a Gaussian distribution with zero mean vector and covariance matrix Σ_s which is a 9×9 diagonal matrix with $\Sigma_{s11} = 4$, $\Sigma_{s22} = 4$, $\Sigma_{s33} = 0.01$, $\Sigma_{s44} = 0.5$, $\Sigma_{s55} = 0.5$, $\Sigma_{s66} = 0.05$, $\Sigma_{s77} = 0.16$, $\Sigma_{s88} = 0.16$, and $\Sigma_{s99} = 2.5$. The random walk covariance matrix Σ_r of $(x, y, z, l, w, h, \gamma, \beta, \alpha)^T$ is also a 9×9 diagonal matrix with $\Sigma_{r11} = 1$, $\Sigma_{r22} = 1$, $\Sigma_{r33} = 0.0009$, $\Sigma_{r44} = 0.01$, $\Sigma_{r55} = 0.01$, $\Sigma_{r66} = 0.0025$, $\Sigma_{r77} = 0.01$, $\Sigma_{r88} = 0.01$, and $\Sigma_{r99} = 0.09$. The RJMCMC move probabilities p_u, p_s, p_d, and p_a are set to be 0.7, 0.1, 0.1, and 0.1, respectively. $\lambda_v = 7$, $\lambda_m = 5$, and $\lambda_d = 0.4$. For RJMCMC sample generation, we set sample number $N = 500$ in the experiment.

4.9 Conclusions

We have presented a sequential Monte Carlo (SMC) framework for variable number of 3D multiple subcellular structure tracking. To detect the subcellular structure, regional maxima of the original volume after applying 3D gray-scale reconstructive opening operation are used to label each object, with the OBV of the dilated regional maxima as the state description for each object. The individual state of each object is then combined into joint state to represent the multiple objects. The marker residual volume appearance model is proposed to detect new objects. RJMCMC moves such update move, identity switch move, disappearing move, and appearing move are applied to generate the samples of the dimension changing joint state

efficiently. Both visual and quantitative experiment results show that our method can track different motion modalities such as Brownian motion and translational motion, and detect object appearing and disappearing correctly.

CHAPTER 5

MERGE AND SPLIT DETECTION IN TRACKING PROTEIN CLUSTERS

5.1 Introduction

The introduction of sensitive electronic imaging devices and the development of methods to tag proteins of interest by green fluorescent protein (GFP) have been the drivers to live-cell study. Understanding the motility of these GFPs is a key to understanding how cells regulate delivery of specific proteins from the site of synthesis to the site of action. At present, biologists either laboriously track a few vesicles by hand, or use commercially available particle tracking programs whose performance is far below expectations for various demands. One GFP image obtained from laser scanning confocal microscope (LSCM) is shown in Fig. 1.1. The green parts of the image are the GFP clusters of interest. Currently, there are two categories of GFP molecule that are under investigation. One is the separated small spot-like GFP molecular particle. Another is the larger GFP molecular cluster, consisting of a number of close contacted spots. Molecular mobility properties of spot-like molecular particles have been investigated a lot [22]. Nonetheless, most methods assume that the spatial configuration of the spot intensity does not change, and will easily fail to handle the GFP molecular clusters. The 2D+T GFP molecular cluster image sequences obtained by LSCM have distinctive properties: 1) variable cluster size; 2) large shape deformation; 3) clusters are close to each other; 4) variable number of objects; 5) diverse motion modalities, such as Brownian motion, and Non-Brownian motion; 6) lacking of distinctive features to discern clusters. These specialties impose

lots of challenges for tracking multiple GFP clusters. Commonly used motion tracking techniques can not be well applied. For instance, template matching method works only when the object of interest has no large deformation and is in a sparse object density environment without high noise. Large inter-frame displacement will mislead the curve/surface evolution of the active contour/surface-based tracking method [11].

Furthermore, there are a few literatures on one of the challenging problems in multiple visual object tracking, which is the splitting and merging of visual objects. Under the assumption of slow inter-frame object movement, Yang *et al.* [67] use Euclidean distance to detect such interaction between objects. Virtual measurements use the split and merge functions to resolve the measurement conflict [68]. In [69], the split and merged measurements are represented by a sparse matrix and solved by MCMC based auxiliary particle filter. The basic assumptions are:

1. Objects are almost non-deformable, thus the tracked objects can be well approximated by circles, ellipses or rectangles;
2. Objects have the same size and shape before and after the interaction.

Unfortunately, GFP clusters have large deformation, and different size and shape. Withers *et al.* [70] tackle this situation using the region overlap information between the consecutive image frames. However, it is limited to slow cell movement.

On the other hand, one of the difficulties in multiple object tracking (MOT) is data association which establishes the connections between the tracks and observations. Two famous algorithms for data association are multiple-hypothesis tracking (MHT) and joint probabilistic data association (JPDA) [50]. MHT has the problem of exponential computation complexity. JPDA is limited to the fixed number of targets and cannot initialize or delete target. Recent literatures successfully treat data association as NP-hard combinatorial problem, and approximate the optimal solution by Markov chain Monte Carlo (MCMC) method [71], [72]. In this chapter, we propose a

novel multiple GFP clusters split and merge tracking framework combined with asymmetric region matching strategy. To save the possible high exponential computation cost, we adapt Markov chain Monte Carlo data association (MCMCDA) method into our framework to find the approximate optimal solution. The method is free of the size and movement of GFP cluster.

The remainder of the chapter is organized as follows. We model the GFP clusters splitting and merging by the asymmetric distance between regions and the region tracking by bipartite graph in Section 5.2 and Section 5.3, respectively. The MCMCDA method is introduced in Section 5.4. In Section 5.5, we present the experiment results. Finally, conclusions and future work are discussed in Section 5.6.

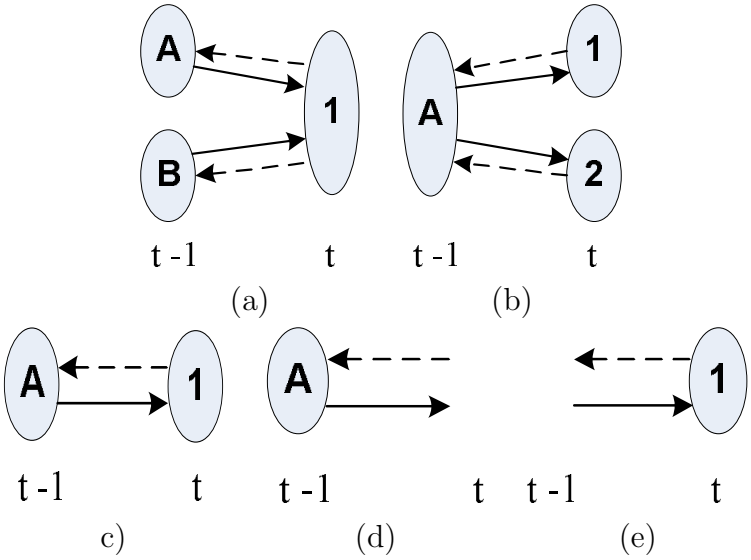


Figure 5.1. Object matching scenarios. (a) is object merging. (b) is object splitting. (c) is object equal match. (d) is object disappearing. (e) is object appearing.

5.2 Modeling of Clusters Split and Merge

5.2.1 Properties of Asymmetric Distance

The interaction between multiple protein clusters is modeled as region splitting and merging. It is assumed that one region will not split and merge with other regions at the same time. Before defining the asymmetric distance, we suppose there is an asymmetric region distance measure $AD(R_1, R_2) \in [0, 1]$ between two regions R_1 and R_2, having the property that $AD(R_1, R_2) \neq AD(R_2, R_1)$ in general. Representing regions in frame t–1 and t by R^i_{t-1}, $i = 1, \cdots, n_{t-1}$ and R^j_t, $j = 1, \cdots, n_t$, respectively, we define forward asymmetric distance (FAD) as $FAD(R^i_{t-1}, R^j_t) = AD(R^i_{t-1}, R^j_t)$ and backward asymmetric distance (BAD) as $BAD(R^i_{t-1}, R^j_t) = AD(R^j_t, R^i_{t-1})$. Different object matching scenarios between regions of consecutive frames are shown in Fig. 5.1, with FAD represented by the solid arrow lines and BAD represented by the dashed arrow lines. The introduction of BAD will facilitate the detection of object splitting.

Ideally, the asymmetric distance should have the *Strict Asymmetric Distance Properties* as follows:

1. $FAD < BAD$, when objects merge;
2. $FAD > BAD$, when object splits;
3. $FAD = BAD$, when object is equal match;
4. $FAD = 1 \wedge BAD = \emptyset$, when object disappear;
5. $FAD = \emptyset \wedge BAD = 1$, when object appear;

Here, $BAD = \emptyset$ means that BAD does not exist, and the same to $FAD = \emptyset$. Due to the uncertainties in the system and observation models, strict asymmetric distance properties do not always hold. Therefore, validation threshold γ_v and scale threshold γ_s are introduced to relax the conditions with the *Relaxed Asymmetric Distance Properties* as:

1. Objects merge: $FAD \leq \gamma_v \wedge BAD > \gamma_v \wedge BAD/FAD > \gamma_s$;
2. Object splits: $FAD > \gamma_v \wedge BAD \leq \gamma_v \wedge FAD/BAD > \gamma_s$;
3. Object equal match:
   (a) $FAD \leq \gamma_v \wedge BAD > \gamma_v \wedge BAD/FAD \leq \gamma_s$;
   (b) $FAD > \gamma_v \wedge BAD \leq \gamma_v \wedge FAD/BAD \leq \gamma_s$;
   (c) $FAD \leq \gamma_v \wedge BAD \leq \gamma_v$;
4. New object appear: $FAD > \gamma_v \wedge BAD > \gamma_v$ for R_t^j;
5. Existing object disappear: $FAD > \gamma_v \wedge BAD > \gamma_v$ for R_{t-1}^i;

In the zero denominator case, the quotient is set to infinity.

Combining *FAD* and *BAD*, the distance between two regions in different consecutive frames is defined as:

$$D(R_{t-1}^i, R_t^j) = \min[FAD(R_{t-1}^i, R_t^j), BAD(R_{t-1}^i, R_t^j)]. \tag{5.1}$$

In the context of image frame, without loss of generality, $p(R_t^j | R_{t-1}^i)$ is assumed to be a Gaussian distribution as :

$$p(R_t^j | R_{t-1}^i) = K \exp(-\lambda D^2(R_{t-1}^i, R_t^j)), \tag{5.2}$$

where K, λ are constant parameters guarantying that $p(R_t^j | R_{t-1}^i)$ is a distribution.

5.2.2 Definition of Asymmetric Distance

The calculation of the asymmetric distance having the properties mentioned in Section 5.2.1 for the detection of regions splitting and merging is defined as follows. At beginning, we subsample the region R_s and represent it by its sample pixels $\{g_{si} : i = 1, \cdots, N\}$, N is the total number of the sample pixels. Then the correspondent pixels in region R_t are found for each sample pixel of R_s by a greedy searching strategy. First, we determine the correspondent pixel in region R_t by the smallest pixel distance

among all the sample pixels of R_s, and delete the correspondence pixel together with its local support from R_t. Then apply the same procedure to the rest of the sample pixels of R_s. In the case that there is no candidate correspondence pixel in R_t, the distance for that sample pixel of R_s to R_t is set to 1. We give the distance measure between two pixels $g_s \in R_s$ and $g_t \in R_t$ as:

$$d_(g_s, g_t) = \frac{1}{|L|} \sum_{g \in L} \left(1 - \frac{\min(I_k(\vec{g}), I_n(\vec{g} + \vec{g}_t - \vec{g}_s))}{\max(I_k(\vec{g}), I_n(\vec{g} + \vec{g}_t - \vec{g}_s))} \right), \tag{5.3}$$

where L is the local support of g_s, $\vec{g}$ is the coordinate vector of g, $|.|$ is the cardinal operation, I_k and I_n are the image functions at time k, n, respectively, with $R_s \subset I_k$ and $R_t \subset I_n$. In the zero denominator case, the quotient is set to 1. We let $d_M(g_{si}, g_{ti})$ be the distance between the sample pixel g_{si} of R_s and its correspondence g_{ti} in R_t. Finally, the asymmetric distance $AD(R_1, R_2)$ is calculated as:

$$AD(R_1, R_2) = \frac{1}{N} \sum_{i=1}^{N} d_M(g_{si}, g_{ti}). \tag{5.4}$$

5.3 Modeling Region Tracking by Bipartite Graph

5.3.1 Introduction to Data Association

The object tracking problem is often modeled as the state-space model. Let x_t^i be the state of the ith object at time t with dimension m , z_t^i be its observation with dimension n. Under the assumption that all the objects have the same general linear motion model and observation model, we have [73]:

$$x_t^i = F x_{t-1}^i + w, \tag{5.5}$$

$$z_t^i = H x_t^i + v, \tag{5.6}$$

where F is a $m \times m$ matrix, H is a $n \times m$ matrix, w, v are assumed to be independent Gaussian white noises with zero means and covariance Q, R, respectively, the super-

script i of x and z indicates the ith target. Equation (5.5) is system model and Eq. (5.6) is observation model. The predicted mean of x_t^i are $\hat{x}_t^{i-} = Fx_t^i$ with covariance:

$$P_t^{i-} = FP_{t-1}^i F^T + Q. \tag{5.7}$$

And the predicted mean of z_t^i is $\hat{z}_t^{i-} = H\hat{x}_t^{i-}$, with covariance:

$$S_t^{i-} = HP_t^{i-} H^T + R. \tag{5.8}$$

Data association is to determine correspondence between each observation and object at each time t. In most literature, the Mahalanobis distance between a measurement z_t and $\hat{z}_t^{i-}$ is as:

$$[z_t - \hat{z}_t^{i-}]^T (S_t^{i-})^{-1} [z_t - \hat{z}_t^{i-}]. \tag{5.9}$$

Equation (5.9) is used to form the validation gate to exclude associations with low probabilities. it works well in desired situation. Difficulties come out when following situations happen: 1) observation may be contended by the close neighbored objects; 2) object appear or disappear; 3) objects merge or split. These situations are very common in 2D GFP clusters tracking, because of its characteristics mentioned in Section 5.1.

5.3.2 Region Tracking by Bipartite Graph

In the 2D GFP clusters tracking problem, the objects are moving in an unmanipulated manner. As the number of tracked clusters growing, a nature way to generalize their motion models is to use a Gaussian diffusion process. When the state variable consists of the horizontal and vertical coordinates, the spatial support of observation validation gate of an object at time t can be enclosed by a 2D circle centered at the region at $t-1$ with the maximum possible object displacement in consecutive frames as its radius. This approximation is valid in our case because the movement

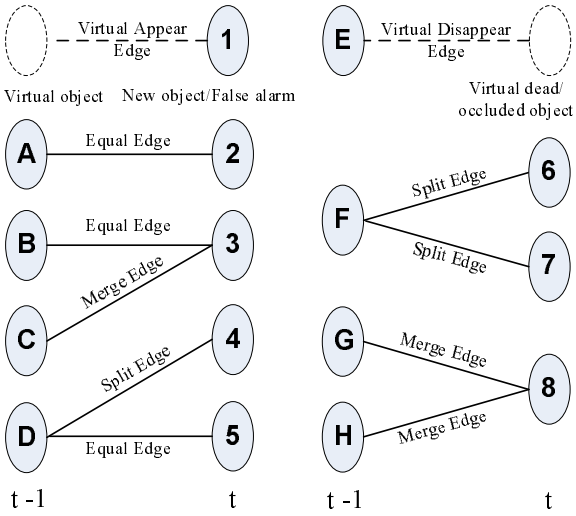


Figure 5.2.Edge types and combinations.

speed of the GFP clusters is bounded. Therefore, The data association problem in region tracking is essentially the region matching problem discussed in Section 5.2. By treating each region as a vertex, the result of connecting all the possible matched regions by edges forms a bipartite graph $G = (R_{t-1}, R_t, E)$, where $R_{t-1} = \cup R_{t-1}^i$, $i = 1, \cdots, n_{t-1}$, and $R_t = \cup R_t^j$, $j = 1, \cdots, n_t$. n_t is the number of regions in the frame at time t, E is the set of all possible edges $e(R_{t-1}^i, R_t^j)$ between R_{t-1}^i and R_t^j with $D(R_{t-1}^i, R_t^j) \leq \gamma_v$.

To model object appearing and disappearing, imagining that: 1) the new region or false alarm are produced by a virtual object in the previous frame; 2) the dead region or occluded region has a virtual object at current frame. Thus, virtual objects R_{t-1}^0 and R_t^0 are introduced at time $t-1$ and t, respectively to reflect this idea. Applying the *Relaxed Asymmetric Distance Properties*, we classify the edges connecting the bipartite graph as:

- Virtual appear edge, e_a;
- Virtual disappear edge, e_d;

- Equal edge, e_e;
- Merge edge, e_m;
- Split edge, e_s.

The illustration for each edge type and its combination is in Fig. 5.2. A region will automatically be assigned an edge of e_a or e_d if it has no any real matching edge such as e_e, e_m, or e_s. The edge set of a vertex can be represented by:

$$V_k = V_a \cup V_d \cup V_e \cup V_m \cup V_s, \tag{5.10}$$

where $V_a = \{e_{ai} : i = 0, \cdots, n_a\}$, $V_d = \{e_{di} : i = 0, \cdots, n_d\}$, $V_e = \{e_{ei} : i = 0, \cdots, n_e\}$, $V_m = \{e_{mi} : i = 0, \cdots, n_m\}$, and $V_s = \{e_{si} : i = 0, \cdots, n_s\}$, n_a, n_d, n_e, n_m, n_s are the degrees of V_a, V_d, V_e, V_m, V_s, respectively. A feasible joint edge association $\theta = \cup V_k$, $k = 1, \cdots, N_v$, where N_v is the total number of non-virtual objects in the two consecutive frames, is defined such that all its real vertices satisfy the following criteria:

1. $|V_a| + |V_d| + |V_e| \leq 1$, for real vertex.
2. $|V_m| * |V_a| = 0$ and $|V_m| * |V_d| = 0$, for real vertex.
3. $|V_s| * |V_a| = 0$ and $|V_s| * |V_d| = 0$, for real vertex.
4. $|V_e| * |V_m| = 0$, for R_{t-1}^i, $i = 1, \cdots, n_{t-1}$.
5. $|V_e| * |V_s| = 0$, for R_t^j, $j = 1, \cdots, n_t$.
6. $|V_m| * |V_s| = 0$, for real vertex.

Here, $|.|$ is the cardinality operation. The last three criteria stipulate that one region can not split and merge at the same time. Therefore ambiguities caused by possible combination of unconstrained regions splitting and merging are avoided. Since one vertex can have the combination of equal edge with split or merge edges, the common scenarios in tracking multiple GFP clusters when one large cluster merges with several small clusters or splits into several small clusters can be represented by the feasible

joint edge association θ. Also, vertex R_{t-1}^0 only has virtual appear edge type e_a, and vertex R_t^0 only has virtual disappear edge type e_d. Now, let $R_{1:t} = \{R_1, \cdots, R_t\}$. Using Bayes' theorem, the posterior of θ can be represented as:

$$\pi(\theta|R_{1:t}) = \frac{1}{C} P(\theta|R_{1:t-1}) p(R_t|\theta, R_{1:t-1}), \tag{5.11}$$

where C is a normalizing constant to keep $\pi(\theta|R_{1:t})$ a distribution. For an image sequence, the effect of $R_{1:t-1}$ can be blended as a reference region R_r, and θ does not depend on $R_{1:t-1}$. Applied to Eq. (5.11), yields:

$$\begin{aligned} \pi(\theta|R_{1:t}) &= \frac{1}{C} P(\theta) p(R_t|\theta, R_r) \\ &= \frac{1}{C} P(\theta) \prod_{(i,j)} p(R_t^j|R_r^i) \prod_{|R_{t-1}^0|} p_a \prod_{|R_t^0|} p_d, \end{aligned} \tag{5.12}$$

where $\{(i,j) : i = 1, \cdots, n_{t-1}, j = 1, \cdots, n_t, e(R_{t-1}^i, R_t^j) \in \theta\}$, R_r^i is the reference region of $R_{1:t-1}^i$, $|R_{t-1}^0|$ and $|R_t^0|$ are degrees of R_{t-1}^0 and R_t^0, respectively, p_a is the likelihood of the appearing edge in the image sequence, and p_d is for the disappearing edge. The prior of θ is defined to be a penalty term for real vertices having degree larger than one and favor the simple joint edge association having less merging and splitting, with expression as:

$$P(\theta) = \exp\Big(\sum_{R \in (R_{t-1} \cup R_t)} \xi \cdot (|R| - 1) \Big), \tag{5.13}$$

where ξ is the constant controlling the penalty effect.

When $|(R_{t-1} \cup R_t)|$ is small, the optimal solution of Eq. 5.12 which maximizes $\pi(\theta|R_{1:t})$ can be solved by traversing the whole solution space which has a size of $2^{|(R_{t-1} \cup R_t)|}$. Nonetheless, the computational cost raises exponentially as $|(R_{t-1} \cup R_t)|$ increase.

5.4 Markov Chain Monte Carlo Data Association

To deal with the exponential computational cost of our method, we introduce Markov chain Monte Carlo data association (MCMCDA) to approximate the optimal solution of Eq. 5.12.

Markov chain Monte Carlo (MCMC) method is essentially the Monte Carlo integration using Markov chain which constructs a sequence of samples with transition kernel $P(.|.)$. A popular MCMC method is Metropolis-Hastings (MH) algorithm, which provides a method to construct a Markov chain such that its stationary distribution $\phi(.)$ is the distribution of interest $\pi(.)$. The idea is simple. At the beginning, randomly pick a state θ. Then sample a state θ' from the proposal distribution $q(.|\theta)$ and accept it with acceptance ratio :

$$\alpha(\theta, \theta') = \min\Big[1, \frac{\pi(\theta')q(\theta|\theta')}{\pi(\theta)q(\theta'|\theta)}\Big]. \tag{5.14}$$

If θ' is accepted, set $\theta = \theta'$. Otherwise, keep the old θ. Readers please refer [60] for details.

Let the collection of all the feasible joint edge association θ be Ω, Metropolis-Hastings (MH) algorithm [60] is used to traverse the set Ω and find its stationary distribution $\pi(\theta|R_{1:t})$. Applying the proposal distribution which selects the edge from E uniformly at random, the acceptance ratio is simplified to:

$$\alpha(\theta, \theta') = \min\Big[1, \frac{\pi(\theta'|R_{1:t})}{\pi(\theta|R_{1:t})}\Big]. \tag{5.15}$$

The algorithm of the MCMCDA at time t is summarized as follows:

1. Select a θ randomly.
2. Select an $e(R^i_{t-1}, R^j_t) \in E$ uniformly at random.
3. Deletion Move: If $e(R^i_{t-1}, R^j_t) \in \theta$, then $\theta' = \theta \setminus e(R^i_{t-1}, R^j_t)$. Go to step 6.
4. Switch Move:

(a) If $e(R_{t-1}^i, R_t^j) \notin \theta$, $e(R_{t-1}^i, R_t^j) = e_e$, and both region R_{t-1}^i and R_t^j each have equal edges $e(R_{t-1}^i, R_t^k)$, $e(R_{t-1}^l, R_t^j)$, respectively, and also $e(R_{t-1}^l, R_t^k)$ is an equal edge, $\theta' = \theta \cup e(R_{t-1}^i, R_t^j) \cup e(R_{t-1}^l, R_t^k) \setminus e(R_{t-1}^i, R_t^k) \setminus e(R_{t-1}^l, R_t^j)$. Go to step 6.

(b) If $e(R_{t-1}^i, R_t^j) \notin \theta$, $e(R_{t-1}^i, R_t^j) = e_e$, and only one region R_{t-1}^i or R_t^j has an equal edge e'_e, $\theta' = \theta \cup e(R_{t-1}^i, R_t^j) \setminus e'_e$. Go to step 6.

5. Addition Move: $\theta' = \theta \cup e(R_{t-1}^i, R_t^j)$.
6. If θ' violates the feasible joint edge association criteria, go to step 2. Otherwise, accept $\theta = \theta'$ with probability $\alpha(\theta, \theta')$ using Eq. (5.15).
7. If the number of samples is less than N, go to step 2.

Here the switch move is added intentionally to accelerate sample mixing rate between different modes of $\pi(\theta|R_{1:t})$.

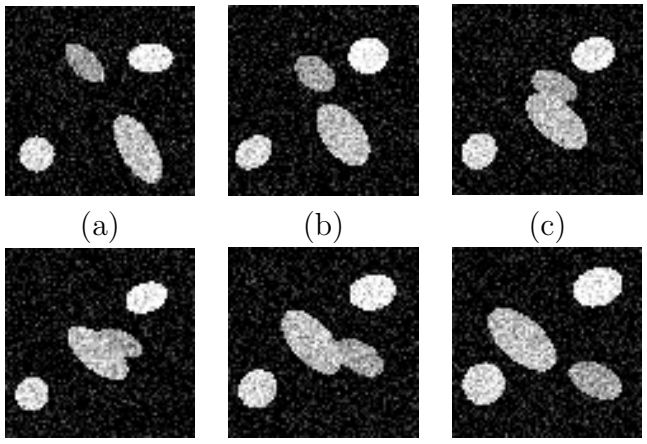


Figure 5.3. Original synthetic images. (a) to (f) are image frames with number 1 to 6.

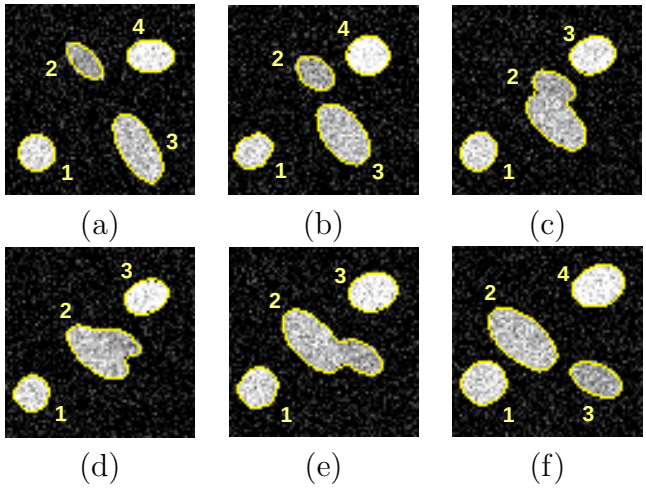


Figure 5.4. Tracking results of synthetic clusters merging and splitting images. (a) to (f) are image frames from 1 to 6.

Table 5.1. FAD and BAD of synthetic frame 1 to 2

Cluster	2(1)	2(2)	2(3)	2(4)
1(1)	(0.1857, 0.1893)	(0.6804, 0.6381)	(0.5509, 0.7031)	(1.0000, 1.0000)
1(2)	(0.8910, 0.9424)	(0.1625, 0.1666)	(0.1952, 0.5469)	(0.3253, 0.4303)
1(3)	(1.0000, 1.0000)	(0.5798, 0.1985)	(0.1495, 0.1395)	(0.7973, 0.6287)
1(4)	(1.0000, 1.0000)	(0.3715, 0.2941)	(0.3912, 0.6379)	(0.1614, 0.1587)

Table 5.2. FAD and BAD of synthetic frame 2 to 3

Cluster	3(1)	3(2)	3(3)
2(1)	(0.1550, 0.1772)	(0.3033, 0.6784)	(1.0000,1.0000)
2(2)	(0.6064, 0.6313)	(0.1385, 0.4694)	(0.2958,0.3140)
2(3)	(0.7885, 0.6276)	(0.1337, 0.1494)	(0.6241,0.3725)
2(4)	(1.0000, 1.0000)	(0.2544, 0.4281)	(0.1498,0.1250)

Table 5.3. FAD and BAD of synthetic frame 3 to 4

Cluster	4(1)	4(2)	4(3)
3(1)	(0.1718, 0.2155)	(0.2067, 0.5580)	(1.0000, 1.0000)
3(2)	(0.6779, 0.3023)	(0.1388, 0.1391)	(0.5214, 0.2737)
3(3)	(1.0000, 1.0000)	(0.2146, 0.3520)	(0.1259, 0.1460)

Table 5.4. FAD and BAD of synthetic frame 4 to 5

Cluster	5(1)	5(2)	5(3)
4(1)	(0.2060, 0.1735)	(0.2318, 0.6306)	(1.0000, 1.0000)
4(2)	(0.4702, 0.2339)	(0.1349, 0.1556)	(0.3099, 0.2102)
4(3)	(1.0000, 1.0000)	(0.2425, 0.5433)	(0.1603, 0.1435)

Table 5.5. FAD and BAD of synthetic frame 5 to 6

Cluster	6(1)	6(2)	6(3)	6(4)
5(1)	(0.1731, 0.1578)	(0.1930, 0.3176)	(1.0000, 1.0000)	(1.0000, 1.0000)
5(2)	(0.4714, 0.1736)	(0.1592, 0.1409)	(0.4248, 0.1412)	(0.4620, 0.2840)
5(3)	(1.0000, 1.0000)	(0.2897, 0.5499)	(0.8720, 0.8350)	(0.1254, 0.1367)

Table 5.6. Edge table of synthetic frame 1 to 2

Cluster	2(1)	2(2)	2(3)	2(4)
1(1)	$*e_e$	n/a	n/a	n/a
1(2)	n/a	$*e_e$	e_m	n/a
1(3)	n/a	e_s	$*e_e$	n/a
1(4)	n/a	e_e	n/a	$*e_e$

Table 5.7. Edge table of synthetic frame 2 to 3

Cluster	3(1)	3(2)	3(3)
2(1)	$*e_e$	n/a	n/a
2(2)	n/a	$*e_m$	e_e
2(3)	n/a	$*e_e$	n/a
2(4)	n/a	e_m	$*e_e$

Table 5.8. Edge table of synthetic frame 3 to 4

Cluster	4(1)	4(2)	4(3)
3(1)	$*e_e$	e_m	n/a
3(2)	n/a	$*e_e$	e_s
3(3)	n/a	e_m	$*e_e$

Table 5.9. Edge table of synthetic frame 4 to 5

Cluster	5(1)	5(2)	5(3)
4(1)	$*e_e$	e_m	n/a
4(2)	e_s	$*e_e$	e_s
4(3)	n/a	e_m	$*e_e$

Table 5.10. Edge table of synthetic frame 5 to 6

Cluster	6(1)	6(2)	6(3)	6(4)
5(1)	$*e_e$	e_m	n/a	n/a
5(2)	e_s	$*e_e$	$*e_s$	e_s
5(3)	n/a	e_m	n/a	$*e_e$

5.5 Experiment Results

To evaluate the performance of our MCMCDA algorithm on detecting cluster splitting and merging, we use both synthetic and real clusters splitting and merging video sequences. The synthetic video sequence is to verify the expected performance

of our method, while its ability to solve the real problem is demonstrated by the noisy real video sequence.

5.5.1 Experiments on Synthetic Clusters Splitting and Merging Video Sequence

To simulate clusters splitting and merging phenomena, we produce four clusters in the video sequence. All clusters are approximated by eclipses with state parameter $x = [u, v, \nu_u, \nu_v, a_l, a_s, \theta]$, where u and v are the coordinate of the center of eclipse, ν_u and ν_v are velocity in u and v axes directions, a_l and a_s are the lengths of the semimajor and semiminor axes, respectively, θ represent the angle between the u axis and the semimajor axis. As shown in Fig. 5.4(a), the labeled clusters 1 and 4 are doing Brownian motion, and the other two clusters are doing translational motion. The system equation uses Eq. (5.5) with constant speed model, where

$$F = \begin{bmatrix} 1 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \tag{5.16}$$

The noise parameter w is defined to be a 7×7 diagonal matrix as:

$$w = \begin{bmatrix} 2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 2 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 4 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 4 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0.7 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0.01 \end{bmatrix}. \tag{5.17}$$

The initial values for each object are set to $x_0^1 = [56, 10,\ 0, 0, 7, 7, pi/6]$, $x_0^2 = [20, 29,\ 5, 5, 10, 5, pi/4]$, $x_0^3 = [54, 50, -5,\ -5, 15, 7, pi/6]$, and $x_0^4 = [18, 55,\ 0, 0, 9, 6, pi/2]$. The intensity of the image background is set to be zero, and the intensities of each cluster are set to be 220, 160, 190, and 250 accordingly. Gaussian additive noise with standard deviation $\sigma = 36$ is applied to each video frame for the simulation of the intensity disturbance. Here we use $x(y)$ to refer to cluster y in frame x. This convention is also used in the rest of this chapter. The whole scenario of cluster merging and splitting is as follows: At frame 1, we detected four clusters, 1(1), 1(2), 1(3), and 1(4), by intensity thresholding method. The four clusters are moving according to their initial values, thus 1(1) and 1(4) are doing Brownian motion, and 1(2) and 1(3) are doing translational motion. At frame 3, clusters 2(2) and 2(3) merge together as cluster 3(2), and continue to be merged as clusters 4(2) and 5(2). At frame 6, cluster 5(2) splits into clusters 6(2) and 6(3). Our MCMCDA algorithm can track the whole cluster merging and splitting process correctly, at the same time without losing the track of clusters 1(1) and 1(4), which have no interaction with others. The asymmetric distances FAD and BAD between consecutive frames from 1 to 6 are shown in Tables 5.1, 5.2, 5.3, 5.4, and 5.5, where the distance is of the format as (FAD, BAD). The edge sets are represented by edge tables in Tables 5.6, 5.7, 5.8,

5.9, and 5.10, with each of the final selected edges indicated by an asterisk in the front. As can been seen in Table 5.7 the merging of cluster 2(2) with cluster 2(3) is represented by a merge edge between clusters 2(2) and 3(2) and an equal edge between clusters 2(3) an 3(2), which can be interpreted as cluster 2(2) attached to cluster 2(3). Similarly in Table 5.10, the splitting of cluster 5(2) is represented by one equal edge between clusters 5(2) and 6(2) and a split edge between clusters 5(2) and 6(3). This means that cluster 6(3) detaches from 6(2). This interpretation of the selected edge configuration can be visually verified in Fig. 5.4.

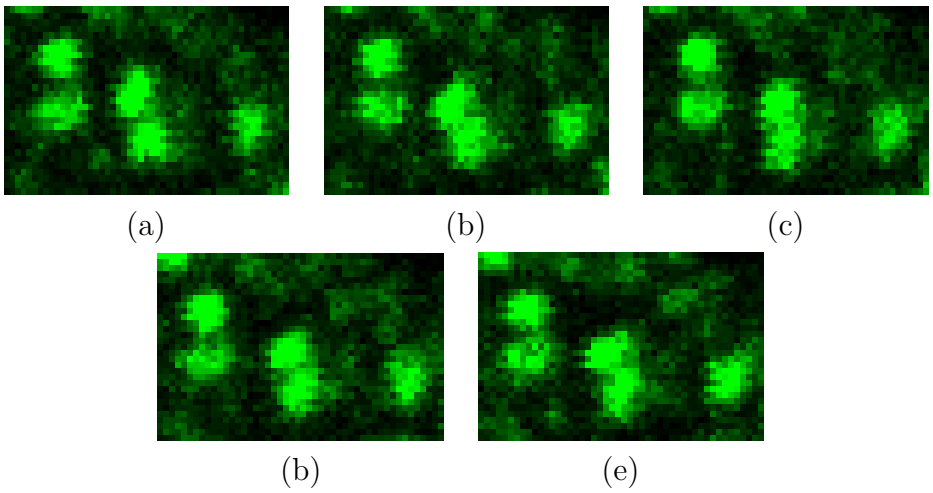


Figure 5.5. Original real images. (a) to (e) are image frames with number 79 to 83.

5.5.2 Experiments on Real Clusters Splitting and Merging Video Sequence

We test our merge and split detection algorithm by using an image sequence acquired by Leica TCS-SP laser scanning confocal microscope with image size of 512×512. The marker guided watershed segmentation method is applied for object detection. Validation and scale thresholds are set to $\gamma_v = 0.24$ and $\gamma_s = 1.3$, respec-

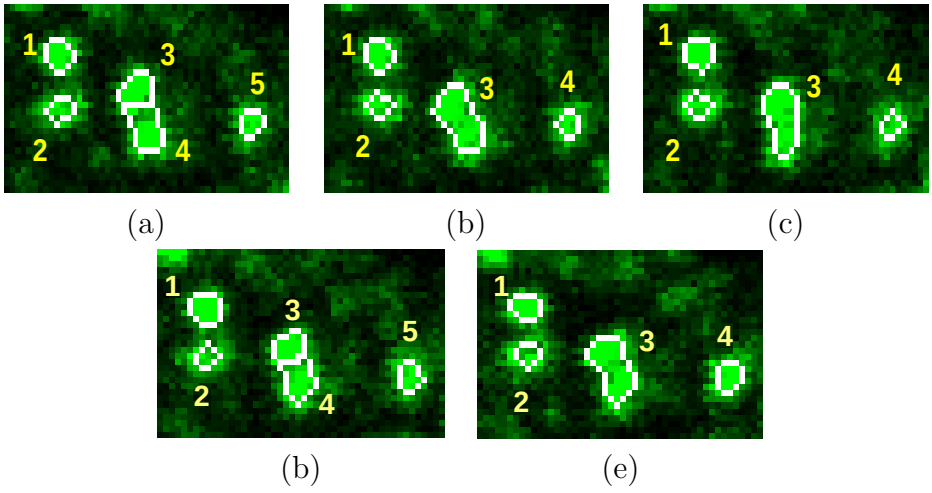


Figure 5.6. Molecular clusters merging and splitting images. (a) to (e) are image frames with number 79 to 83.

Table 5.11. FAD and BAD of frame 79 to 80

Cluster	80(1)	80(2)	80(3)	80(4)
79(1)	(0.2200, 0.1721)	(0.2116, 0.2377)	(0.1789, 0.4228)	(1.0000, 0.6841)
79(2)	(0.2381, 0.3144)	(0.1503, 0.2050)	(0.1742, 0.6076)	(0.6431, 0.5918)
79(3)	(0.2450, 0.2625)	(0.2559, 0.2946)	(0.1771, 0.3608)	(0.2947, 0.2248)
79(4)	(0.2655, 0.2592)	(0.1909, 0.2033)	(0.1599, 0.2108)	(0.3658, 0.1571)
79(5)	(0.8354, 1.0000)	(0.6212, 0.8934)	(0.1848, 0.5929)	(0.1612, 0.1718)

Table 5.12. FAD and BAD of frame 80 to 81

Cluster	81(1)	81(2)	81(3)	81(4)
80(1)	(0.2000, 0.1506)	(0.2590, 0.2184)	(0.1952, 0.3833)	(1.0000, 0.6637)
80(2)	(0.3021, 0.1869)	(0.1732, 0.1656)	(0.2005, 0.2703)	(0.8954, 0.5733)
80(3)	(0.5416, 0.1731)	(0.3100, 0.1795)	(0.1745, 0.1740)	(0.4849, 0.1672)
80(4)	(0.6894, 1.0000)	(0.6156, 0.8633)	(0.1676, 0.3880)	(0.1608, 0.1694)

Table 5.13. FAD and BAD of frame 81 to 82

Cluster	82(1)	82(2)	82(3)	82(4)	82(5)
81(1)	(0.1471, 0.1433)	(0.2490, 0.2632)	(0.1772, 0.1804)	(0.2168, 0.3584)	(1.0000, 0.6862)
81(2)	(0.2685, 0.2385)	(0.1598, 0.1527)	(0.2460, 0.2582)	(0.1977, 0.1975)	(0.8644, 0.6011)
81(3)	(0.4723, 0.1894)	(0.2534, 0.2041)	(0.4530, 0.1848)	(0.2310, 0.1846)	(0.3713, 0.1603)
81(4)	(0.6993, 0.8583)	(0.6120, 0.8789)	(0.2549, 0.2981)	(0.1675, 0.3396)	(0.1295, 0.1608)

Table 5.14. FAD and BAD of frame 82 to 83

Cluster	83(1)	83(2)	83(3)	83(4)
82(1)	(0.1580, 0.1715)	(0.2424, 0.2923)	(0.1716, 0.4342)	(0.7119, 0.7409)
82(2)	(0.2552, 0.2400)	(0.1934, 0.1899)	(0.1863, 0.2763)	(0.6598, 0.5985)
82(3)	(0.2029, 0.2099)	(0.2433, 0.2481)	(0.1536, 0.4383)	(0.2219, 0.2614)
82(4)	(0.2993, 0.2342)	(0.2042, 0.1945)	(0.1597, 0.2313)	(0.2157, 0.2017)
82(5)	(0.6643, 1.0000)	(0.6153, 0.7599)	(0.1491, 0.4308)	(0.1690, 0.1644)

Table 5.15. Edge table of frame 79 to 80

Cluster	80(1)	80(2)	80(3)	80(4)
79(1)	$*e_e$	e_e	e_m	n/a
79(2)	e_m	$*e_e$	e_m	n/a
79(3)	e_e	e_e	$*e_m$	e_e
79(4)	e_e	e_e	$*e_e$	e_s
79(5)	n/a	n/a	e_m	$*e_e$

Table 5.16. Edge table of frame 80 to 81

Cluster	81(1)	81(2)	81(3)	81(4)
80(1)	$*e_e$	e_e	e_m	n/a
80(2)	e_s	$*e_e$	e_e	n/a
80(3)	e_s	e_s	$*e_e$	e_s
80(4)	n/a	n/a	e_m	$*e_e$

Table 5.17. Edge table of frame 81 to 82

Cluster	82(1)	82(2)	82(3)	82(4)	82(5)
81(1)	$*e_e$	e_e	e_e	e_m	n/a
81(2)	e_e	$*e_e$	e_e	e_e	n/a
81(3)	e_s	e_e	$*e_s$	$*e_e$	e_s
81(4)	n/a	n/a	e_e	e_m	$*e_e$

Table 5.18. Edge table of frame 82 to 83

Cluster	83(1)	83(2)	83(3)	83(4)
82(1)	$*e_e$	e_e	e_m	n/a
82(2)	e_e	$*e_e$	e_e	n/a
82(3)	e_e	e_e	$*e_m$	e_e
82(4)	e_e	e_e	$*e_e$	e_e
82(5)	n/a	n/a	e_m	$*e_e$

tively. Here, we present portions of frame 79 to 83 here, depicting that two molecular clusters merge and split as shown in Fig. 5.5, The phenomena is that clusters 3 and 4 in frame 79 merge together into cluster 3 in frame 80, and continue to be merged in frame 81 as cluster 3. At frame 82, it splits into clusters 3 and 4, and merge again in frame 83. At the same time, other clusters have no interactions. Although all the clusters do not have large movement in this case, it is enough to show the success of our method since the speed information is not considered in the calculation of asymmetric distance.

The asymmetric distances FAD and BAD between consecutive frames from 79 to 83 are shown in Tables 5.11, 5.12, 5.13, and 5.14. The edge sets are represented by edge tables in Tables 5.15, 5.16, 5.17, and 5.18, where each final selected edge is indicated by an asterisk in the front. The merging process of clusters 79(3) and 79(4) is represented by one merge edge and one equal edge as shown in Table 5.15, which can be interpreted as that cluster 79(3) attached to cluster 79(4) and the merged result is cluster 80(3). For splitting process, cluster 81(3) is indicated by having a split edge and an equal edge as shown in Table 5.17, which can be interpreted as that cluster 81(3) splits into cluster 82(3) and 82(4). For the edge table between frames 79 and 80, there are a total of 16 edges there, the solution space is $2^{16} = 65536$. The MCMCDA method is applied to find the approximate optimal solution, though it turns out to be the optimal one.

It worth noticing that there is a tendency to establish the wrong edges between clusters such as: 79(4) to 80(4), 80(3) to 81(4), 81(3) to 82(5), 82(2) to 83(3), and 82(5) to 83(3) by simply evaluating the distance value D between them using Eq.(5.1). It is correctly avoided by our method.

5.6 Conclusions and Future Work

In this chapter, we propose a novel multiple GFP clusters split and merge tracking framework combined with asymmetric region matching strategy and MCMCDA approximate method to find the optimal solution efficiently. It can not only track individual clusters and the interactions between them, but also provide meaningful interpretation about the merging and splitting interactions such as object attaching and detaching.

In the experiment, it is also noticed that due to the noisiness of the GFP video sequence, the calculation of the asymmetric distance is not always robust to pro-

vide correct edge mapping information. A more robust asymmetric distance measure combined with other cluster information such as orientation and shape can be used to improve its performance. However, the current algorithm provides a novel framework for tracking clusters with interaction. Thus for the future work, we are going to improve the robustness of the algorithm under different scenarios, especially when the cluster intensity is unstable between adjacent frames in real situation. Adaptive thresholds for validation and scale, and the connection between the broken trajectories and the complete occlusions, will also be considered in the future work.

APPENDIX A

CALCULATION OF JACOBIAN TERMS OF RJMCMC MOVES

In this appendix, we present the calculation of the Jacobian terms of the RJMCMC moves.

A.1 Jacobian Calculation

Supposing we have a k-dimension variable $\mathbf{x} = [x_1, x_2, \cdots, x_k]$, $\mathbf{x} \in \mathbb{R}^k$, there is a function $f : \mathbb{R}^k \to \mathbb{R}^k$, mapping $\mathbf{x}$ to a k-dimension variable $\mathbf{y} = f(\mathbf{x})$, in detail $\mathbf{y} = [f_1(\mathbf{x}), f_2(\mathbf{x}), \cdots, f_k(\mathbf{x})]$, $\mathbf{y} \in \mathbb{R}^k$. The Jacobian matrix of function f is denoted by $J_f(\mathbf{x})$. Its calculation is:

$$J_f(\mathbf{x}) = \frac{\partial \mathbf{y}}{\partial \mathbf{x}} = \begin{bmatrix} \frac{\partial f_1(\mathbf{X})}{\partial x_1} & \frac{\partial f_1(\mathbf{X})}{\partial x_2} & \cdots & \frac{\partial f_1(\mathbf{X})}{\partial x_k} \\ \frac{\partial f_2(\mathbf{X})}{\partial x_1} & \frac{\partial f_2(\mathbf{X})}{\partial x_2} & \cdots & \frac{\partial f_2(\mathbf{X})}{\partial x_k} \\ \vdots & \vdots & \cdots & \vdots \\ \frac{\partial f_k(\mathbf{X})}{\partial x_1} & \frac{\partial f_k(\mathbf{X})}{\partial x_2} & \cdots & \frac{\partial f_k(\mathbf{X})}{\partial x_k} \end{bmatrix}. \tag{A.1}$$

The related Jacobian determinant is: $\det |J_f(\mathbf{x})|$.

A.2 Jacobian Terms of RJMCMC Moves

For the simplicity of the explanation, we suppose there are two objects being tracked before each move, namely $O1$ with state $\mathbf{X}_{n_t.1}$ and $O2$ with state $\mathbf{X}_{n_t.2}$, and let the current joint state be $\mathbf{X}_{n_t} = \{\mathbf{X}_{n_t.1}, \mathbf{X}_{n_t.2}\}$. The dimension of each individual state $\mathbf{X}_{n_t.i}$, $i \in n_t$, is k. So $\mathbf{X}_{n_t.i} \in \mathbb{R}^k$.

A.2.1 Update Move

For update move, we update one object at a time. Suppose we generate the auxiliary vector $\mathbf{U} \in \mathbb{R}^k$ from the proposal to update object $O2$. The dimension matching of the current joint state is $\{\mathbf{X}_{n_t}, \mathbf{U}\}$, in detail $\{\mathbf{X}_{n_t.1}, \mathbf{X}_{n_t.2}, \mathbf{U}\}$. The pro-

posed dimension matching joint state is $\{\mathbf{X}'_{n_t}, \mathbf{U}'\}$, in detail $\{\mathbf{X}'_{n_t.1}, \mathbf{X}'_{n_t.2}, \mathbf{U}'\}$. The mapping relation between the current joint state and the proposal joint state is:

$$\mathbf{X}'_{n_t.1} = \mathbf{X}_{n_t.1}, \tag{A.2}$$

$$\mathbf{X}'_{n_t.2} = \mathbf{X}_{n_t.2} + \mathbf{U}, \tag{A.3}$$

$$\mathbf{U}' = -\mathbf{U}. \tag{A.4}$$

The Jacobian term of RJMCMC update move is:

$$\frac{\partial(\mathbf{X}'_{n_t}, \mathbf{U}')}{\partial(\mathbf{X}_{n_t}, \mathbf{U})} = \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 1 & -1 \end{vmatrix} = -1. \tag{A.5}$$

A.2.2 Appear Move

In this move, we are adding object $O3$ with its state $\mathbf{X}'_{n_t.3}$. Suppose we generate the auxiliary vector $\mathbf{U} \in \mathbb{R}^k$ from the proposal for the object $O3$. The dimension matching of the current joint state is $\{\mathbf{X}_{n_t}, \mathbf{U}\}$, in detail $\{\mathbf{X}_{n_t.1}, \mathbf{X}_{n_t.2}, \mathbf{U}\}$. The proposed dimension matching joint state is $\{\mathbf{X}'_{n_t}\}$, in detail $\{\mathbf{X}'_{n_t.1}, \mathbf{X}'_{n_t.2}, \mathbf{X}'_{n_t.3}\}$. The mapping relation between the current joint state and the proposal joint state is:

$$\mathbf{X}'_{n_t.1} = \mathbf{X}_{n_t.1}, \tag{A.6}$$

$$\mathbf{X}'_{n_t.2} = \mathbf{X}_{n_t.2}, \tag{A.7}$$

$$\mathbf{X}'_{n_t.3} = \mathbf{U}. \tag{A.8}$$

The Jacobian term of RJMCMC appear move is:

$$\frac{\partial(\mathbf{X}'_{n_t})}{\partial(\mathbf{X}_{n_t}, \mathbf{U})} = \begin{vmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{vmatrix} = 1. \tag{A.9}$$

A.2.3 Disappear Move

In this move, we are deleting object $O2$ with its state $\mathbf{X}_{n_t.2}$. The dimension matching of the current joint state is $\{\mathbf{X}_{n_t}\}$, in detail $\{\mathbf{X}_{n_t.1}, \mathbf{X}_{n_t.2}\}$. Suppose we generate the auxiliary vector $\mathbf{U}' \in \mathbb{R}^k$ from the proposal for object $O2$. The proposed dimension matching joint state is $\{\mathbf{X}'_{n_t}, \mathbf{U}'\}$, in detail $\{\mathbf{X}'_{n_t.1}, \mathbf{U}'\}$. The mapping relation between the current joint state and the proposal joint state is:

$$\mathbf{X}'_{n_t.1} = \mathbf{X}_{n_t.1}, \tag{A.10}$$

$$\mathbf{U}' = \mathbf{X}_{n_t.2}. \tag{A.11}$$

The Jacobian term of RJMCMC disappear move is:

$$\frac{\partial(\mathbf{X}'_{n_t}, \mathbf{U}')}{\partial(\mathbf{X}_{n_t})} = \begin{vmatrix} 1 & 0 \\ 0 & 1 \end{vmatrix} = 1. \tag{A.12}$$

A.2.4 Identity Swap Move

In this move, we swap the identities of the two object $O1$ and $O2$. The current joint state is $\{\mathbf{X}_{n_t}\}$, in detail $\{\mathbf{X}_{n_t.1}, \mathbf{X}_{n_t.2}\}$. The proposed joint state is $\{\mathbf{X}'_{n_t}\}$, in detail $\{\mathbf{X}'_{n_t.1}, \mathbf{X}'_{n_t.2}\}$. The mapping relation between the current joint state and the proposal joint state is:

$$\mathbf{X}'_{n_t.1} = \mathbf{X}_{n_t.2}, \tag{A.13}$$

$$\mathbf{X}'_{n_t.2} = \mathbf{X}_{n_t.1}. \tag{A.14}$$

The Jacobian term of RJMCMC identity swap move is:

$$\frac{\partial(\mathbf{X}'_{n_t})}{\partial(\mathbf{X}_{n_t})} = \begin{vmatrix} 0 & 1 \\ 1 & 0 \end{vmatrix} = -1. \tag{A.15}$$

A.2.5 Height Swap Move

In this move, we swap the height z of the two object $O1$ and $O2$. The current joint state is $\{\mathbf{X}_{n_t}\}$, in detail $\{\mathbf{X}_{n_t.1}, \mathbf{X}_{n_t.2}\}$. The proposed joint state is $\{\mathbf{X}'_{n_t}\}$, in detail $\{\mathbf{X}'_{n_t.1}, \mathbf{X}'_{n_t.2}\}$. Since $\mathbf{X}_{t.i} = (l, w, \theta, x, y, z)$, the mapping relation between the current joint state and the proposal joint state is:

$$\xi'_{n_t.1} = \xi_{n_t.1}, \tag{A.16}$$

$$z'_{n_t.1} = z_{n_t.2}, \tag{A.17}$$

$$\xi'_{n_t.2} = \xi_{n_t.2}, \tag{A.18}$$

$$z'_{n_t.2} = z_{n_t.1}, \tag{A.19}$$

where $\xi \in \{l, w, \theta, x, y\}$. The Jacobian term of RJMCMC identity height swap move is:

$$\frac{\partial(\mathbf{X}'_{n_t})}{\partial(\mathbf{X}_{n_t})} = \begin{vmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \end{vmatrix} = -1. \tag{A.20}$$

APPENDIX B

PROOF OF THE MINIMUM/MAXIMUM SUM-OF-ABSOLUTE-DIFFERENCES (SAD) BETWEEN TWO SETS OF EQUAL CARDINALITY

In this appendix, we present our proof of the minimum/maximum sum-of-absolute-differences SAD) between two sets of equal cardinality by using mathematical induction method.

B.1 Problem Formulation

Supposing there are two sets A_n and B_n, with $A_n = \{a_1, \ldots, a_i, \ldots, a_n\}$ and $B_n = \{b_1, \ldots, b_i, \ldots, b_n\}$, $a_i \in \mathbb{R}$, $b_i \in \mathbb{R}$, $n \geq 1$, and F_n is the set of all the bijective mappings from A_n to B_n, we define a sub set of mappings $F_n^* \in F_n$, such that $f_n^* \in F_n^*$ satisfies the following:

$$f_n^* = \arg\min_{f_n \in F_n} V(f_n, A_n, B_n), \tag{B.1}$$

where

$$V(f_n, A_n, B_n) = \sum_{i}^{n} |a_i - f_n(a_i)|. \tag{B.2}$$

B.2 Claim and Ideas

We propose a method to find one of the f_n^*.

Claim: Let's rearrange sets A_n and B_n into monotonously increasing ordered sets A'_n and B'_n, such that $A'_n = \{a'_1, \ldots, a'_i, \ldots, a'_n\}$ and $B'_n = \{b'_1, \ldots, b'_i, \ldots, b'_n\}$, with $a'_1 < a'_2 < \ldots < a'_{n-1} < a'_n$, and $b'_1 < b'_2 < \ldots < b'_{n-1} < b'_n$, and define a bijective mapping f'_n, $b'_i = f'_n(a'_i)$. Then, we say that $f'_n \in F_n^*$.

We use induction method to prove our claim. The idea is as follows. We only need prove that $f'_n \in F_n^*$ for sets A_n and B_n. Assuming we construct sets A_n and B_n from empty set $A_0 = \emptyset$ and $B_0 = \emptyset$, since the order of the construction does not matter, we can, at each time i, $i > 0$, only add a'_i to A_{i-1}, and b'_i to B_{i-1}, till $i = n$.

B.3 Lemmas

Before giving the final proof of our claim, there are two lemmas need to be introduced first.

Lemma 1: If there is a method to divide a set A_n into two disjoint subset A_m and A_{n-m}, $0 < m < n$, with $A_m \cup A_{n-m} = A_n$. Similarly $B_m \cup B_{n-m} = B_n$, then

$$V(f_n^*, A_n, B_n) \leq V(f_m^*, A_m, B_m) + V(f_{n-m}^*, A_{n-m}, B_{n-m}). \tag{B.3}$$

The equality is established when each of the $(a_i, f_n^*(a_i))$ pair is in the same subset pair $\{A_n, B_n\}$ or $\{A_{n-m}, B_{n-m}\}$

Proof: Since $A_m \cup A_{n-m} = A_n$ and $B_m \cup B_{n-m} = B_n$, then the combination of the mapping f_m^* and f_{n-m}^* is just one case in F_n. Based on the definition of f_n^*, we get the inequality established. The proof of equal condition is trivial, since the $(a_i, f_n^*(a_i))$ relation is not changed before or after the separation of the set A_n and B_n.

Lemma 2:

$$V(f_1', A_1, B_1) = V(f_1^*, A_1, B_1) = V(f_1, A_1, B_1). \tag{B.4}$$

Proof: Since there is only one element in A_1 and B_1, the number of available mapping from A_1 to B_1 is one. So $f_1' = f_1^* = f_1 = F_1$.

B.4 Proof

Step 1: Supposing at $i = 1$, we have $A_1 = \{a_1'\}$ and $B_1 = \{b_1'\}$. Then follows the construction method mentioned above, we get $A_2 = \{a_1', a_2'\}$ and $B_2 = \{b_1', b_2'\}$ There are just two mappings in F_2, namely $f_{2.1}$ and $f_{2.2}$. Let $b_1' = f_{2.1}(a_1')$ and $b_2' = f_{2.1}(a_2')$ for $f_{2.1}$, and $b_2' = f_{2.2}(a_1')$ and $b_1' = f_{2.2}(a_2')$ for $f_{2.2}$. There are six possible orders of a_1', a_2', b_1', b_2'.

1. case 1: $b'_1 \le b'_2 \le a'_1 \le a'_2$;
2. case 2: $b'_1 \le a'_1 \le b'_2 \le a'_2$;
3. case 3: $b'_1 \le a'_1 \le a'_2 \le b'_2$;
4. case 4: $a'_1 \le b'_1 \le b'_2 \le a'_2$;
5. case 5: $a'_1 \le b'_1 \le a'_2 \le b'_2$;
6. case 6: $a'_1 \le a'_2 \le b'_1 \le b'_2$.

It is easy to verify that $f'_2 = f_{2.1} \in F_2^*$ in each case.

Step 2: Supposing that the claim for $i = n-2$, $n \ge 3$, is established, that $f'_{n-1} \in F_{n-1}^*$. Then for $i = n-1$, we add a'_n to A_{n-1}, get A_n, and b'_n to B_{n-1} get B_n.

Here we have two cases to deal with. case 1: a'_n is paired with b'_n, then we get the

$$\begin{aligned} V(f_n, A_n, B_n) &= V(f_{n-1}, \{A_n \backslash a'_n\}, \{B_n \backslash b'_n\}) + V(f_1, a'_n, b'_n) \\ &= V(f_{n-1}, \{A_n \backslash a'_n\}, \{B_n \backslash b'_n\}) + V(f_1^*, a'_n, b'_n) && \text{; use Lemma 2} \\ &\ge V(f_{n-1}^*, \{A_n \backslash a'_n\}, \{B_n \backslash b'_n\}) + V(f_1^*, a'_n, b'_n) && \text{; definition of } f^*. \end{aligned}$$

case 2: a'_n is paired with b'_j, and b'_n is paired with a'_i. Use the result in Step 1, we have

$$\begin{aligned} V(f_2, \{a'_i, a'_n\}, \{b'_j, b'_n\}) &\ge V(f_2^*, \{a'_i, a'_n\}, \{b'_j, b'_n\}) && \text{; definition of } f^* \\ &= V(f_1^*, \{a'_i\}, \{b'_j\}) + V(f_1^*, \{a'_n\}, \{b'_n\}) \\ &\quad \text{; use Step 1, for } a'_i < a'_n, b'_j < b'_n. \end{aligned} \tag{B.5}$$

Then use the results above, we get:

$$\begin{aligned} V(f_n, A_n, B_n) &= V(f_{n-2}, \{A_n \backslash \{a'_i, a'_n\}\}, \{B_n \backslash \{b'_j, b'_n\}\}) + V(f_2, \{a'_i, a'_n\}, \{b'_j, b'_n\}) \\ &\ge V(f_{n-2}, \{A_n \backslash \{a'_i, a'_n\}\}, \{B_n \backslash \{b'_j, b'_n\}\}) \\ &\quad + V(f_1^*, \{a'_i\}, \{b'_j\}) + V(f_1^*, \{a'_n\}, \{b'_n\}) \text{ ; use Eq.B.5} \end{aligned}$$

$$
\begin{aligned}
&\geq V(f^*_{n-2}, \{A_n \backslash \{a'_i, a'_n\}\}, \{B_n \backslash \{b'_j, b'_n\}\}) \\
&\quad + V(f^*_1, \{a'_i\}, \{b'_j\}) + V(f^*_1, \{a'_n\}, \{b'_n\}) \text{ ; definition of } f^* \\
&\geq V(f^*_{n-1}, \{A_n \backslash a'_n\}, \{B_n \backslash b'_n\}) + V(f^*_1, a'_n, b'_n) \text{ ; use Lemma 1.}
\end{aligned}
$$

Concluding case 1 and 2, we get:

$$
\begin{aligned}
V(f_n, A_n, B_n) &\geq V(f^*_{n-1}, \{A_n \backslash a'_n\}, \{B_n \backslash b'_n\}) + V(f^*_1, a'_n, b'_n) \\
&= V(f'_{n-1}, \{A_n \backslash a'_n\}, \{B_n \backslash b'_n\}) + V(f^*_1, a'_n, b'_n) \text{ ; assumption of Step 2} \\
&= \quad V(f'_n, A_n, B_n) \text{ ; } a'_n, b'_n \text{ are the largest.}
\end{aligned}
$$

The proof is completed.

Similarly, we can define a subset of mappings $F^*_n \in F_n$, such that $f^*_n \in F^*_n$ satisfies the following:

$$
f^*_n = \arg\max_{f_n \in F_n} V(f_n, A_n, B_n). \tag{B.6}
$$

Then we rearrange sets A_n and B_n into monotonously increasing and decreasing ordered sets A'_n, such that $A'_n = \{a'_1, \ldots, a'_i, \ldots, a'_n\}$ and $B'_n = \{b'_1, \ldots, b'_i, \ldots, b'_n\}$, with $a'_1 < a'_2 < \ldots < a'_{n-1} < a'_n$, and B'_n in monotonously decreasing order $b'_1 > b'_2 > \ldots > b'_{n-1} > b'_n$, and define a bijective mapping f'_n, $b'_i = f'_n(a'_i)$. Then we claim $f'_n \in F^*_n$.

The proof uses the same method applied for minimum mapping, and changes the larger equal sign to less equal sign in Lemma 1 and Step 2.

APPENDIX C

CONVERSION BETWEEN ROTATION MATRIX AND ANGLES

In this section, we present the conversion between rotation matrix and angles.

C.1 Conversion Between Rotation Matrix and Angles

Let $v_1 = [v_{1.x}, v_{1.y}, v_{1.z}]^T$, $v_2 = [v_{2.x}, v_{2.y}, v_{2.z}]^T$, and $v_3 = [v_{3.x}, v_{3.y}, v_{3.z}]^T$, and suppose that the major, medium, minor axes of a 3D object have the same directions as x, y, and z axes originally, to let the object has the new orientation described by v_1, v_2 and v_3, the angles that the object rotates about the $\hat{x}$, $\hat{y}$, and $\hat{z}$ axes, which pass through the center of OBV and parallel to x, y, and z axes are γ, β, α, respectively. Here we suppose that the OBV center is fixed. We use the following formula for the conversion between the eigenvectors and rotation angles.

The rotation matrix around $\hat{x}$ axis is:

$$R_{\hat{x}}(\gamma) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{bmatrix}. \tag{C.1}$$

Similarly,

$$R_{\hat{y}}(\beta) = \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix}, \tag{C.2}$$

and

$$R_{\hat{z}}(\alpha) = \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{C.3}$$

are rotation matrices around $\hat{y}$ and $\hat{z}$ axes.

The total rotation matrix is:

$$R_{\hat{x}\hat{y}\hat{z}}(\gamma, \beta, \alpha) = R_{\hat{z}}(\alpha) R_{\hat{y}}(\beta) R_{\hat{x}}(\gamma)$$

$$
\begin{aligned}
&= \begin{bmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\beta & 0 & \sin\beta \\ 0 & 1 & 0 \\ -\sin\beta & 0 & \cos\beta \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\gamma & -\sin\gamma \\ 0 & \sin\gamma & \cos\gamma \end{bmatrix} \\
&= \begin{bmatrix} \cos\alpha\cos\beta & \cos\alpha\sin\beta\sin\gamma - \sin\alpha\cos\gamma & \cos\alpha\sin\beta\cos\gamma + \sin\alpha\sin\gamma \\ \sin\alpha\cos\beta & \sin\alpha\sin\beta\sin\gamma + \cos\alpha\cos\gamma & \sin\alpha\sin\beta\cos\gamma - \cos\alpha\sin\gamma \\ -\sin\beta & \cos\beta\sin\gamma & \cos\beta\cos\gamma \end{bmatrix}. \quad \text{(C.4)}
\end{aligned}
$$

Let

$$
\begin{bmatrix} \cos\alpha\cos\beta & \cos\alpha\sin\beta\sin\gamma - \sin\alpha\cos\gamma & \cos\alpha\sin\beta\cos\gamma + \sin\alpha\sin\gamma \\ \sin\alpha\cos\beta & \sin\alpha\sin\beta\sin\gamma + \cos\alpha\cos\gamma & \sin\alpha\sin\beta\cos\gamma - \cos\alpha\sin\gamma \\ -\sin\beta & \cos\beta\sin\gamma & \cos\beta\cos\gamma \end{bmatrix} = \begin{bmatrix} r_{1.x} & r_{2.x} & r_{3.x} \\ r_{1.y} & r_{2.y} & r_{3.y} \\ r_{1.z} & r_{2.z} & r_{3.z} \end{bmatrix}, \quad \text{(C.5)}
$$

from Eq. C.5, we can easily convert between rotation angles and rotation matrix. From ration matrix to rotation angles, we get

$$
\beta = \tan_2^{-1}\left(-r_{1.z}, \sqrt{r_{1.x}^2 + r_{1.y}^2}\right), \quad \text{(C.6)}
$$

$$
\alpha = \tan_2^{-1}\left(\frac{r_{1.y}}{\cos\beta}, \frac{r_{1.x}}{\cos\beta}\right), \quad \text{(C.7)}
$$

$$
\gamma = \tan_2^{-1}\left(\frac{r_{2.z}}{\cos\beta}, \frac{r_{3.z}}{\cos\beta}\right), \quad \text{(C.8)}
$$

where $\tan_2^{-1}(y, x)$ is a function returning an angle by computing ordinary $\tan^{-1}(\frac{y}{x})$ and using the x and y signs to determine the quadrant of the resulting angle. Here we only let $-\frac{\pi}{2} \leq \beta \leq -\frac{\pi}{2}$. In the case $\beta = \pm\frac{\pi}{2}$, only $\alpha \pm \gamma$, the sum or the difference of α and γ, can be computed, we set $\alpha = 0.0$ according to convention, and get

$$
\begin{aligned}
\beta &= \frac{\pi}{2}, && \text{(C.9)} \\
\alpha &= 0.0, && \text{(C.10)} \\
\gamma &= \tan_2^{-1}(r_{2.x}, r_{2.y}), && \text{(C.11)}
\end{aligned}
$$

or

$$\beta = -\frac{\pi}{2}, \tag{C.12}$$

$$\alpha = 0.0, \tag{C.13}$$

$$\gamma = -\tan_2^{-1}(r_{2.x}, r_{2.y}). \tag{C.14}$$

REFERENCES

[1] D. Davenport and J. A. C. Nicol, "Luminescence in hydromedusae," *Proceedings of the Royal Society of London. Series B, Biological Sciences*, vol. 144, no. 916, pp. 399–411, Nov. 29 1955.

[2] O. Shimomura, F. H. Johnson, and Y. Saiga, "Extraction, purification and properties of aequorin, a bioluminescent protein from the luminous hydromedusan, aequorea," *Journal of Cellular and Comparative Physiology*, vol. 59, no. 3, pp. 223–239, 1962.

[3] J. Morin and J. Hastings, "Biochemistry of the bioluminescence of colonial hydroids and other coelenterates," *Journal of cellular physiology*, vol. 77, no. 3, pp. 305–312, Jun 1971.

[4] Q. Wen, J. Gao, A. Kosaka, H. Iwaki, K. Luby-Phelps, and D. Mundy, "A particle filter framework using optimal importance function for protein molecules tracking," in *IEEE International Conference on Image Processing*, vol. 1, Genoa, Italy, Sept 2005, pp. 1161–1164.

[5] Q. Wen, J. Gao, and K. Luby-Phelps, "Region based tracking of protein molecules," in *IEEE International Symposium on Biomedical Imaging*, vol. 1, April 6-9 2006, pp. 574–577.

[6] Q. Wen, J. Gao, and K. Luby-Phelps, "Feature selection, matching, and evaluation for subcellular structure tracking," in *IEEE 2006 International Conference of the Engineering in Medicine and Biology Society*, August 2006, pp. 3013–3016.

[7] Q. Wen and J. Gao, "Tracking interacting subcellular structures by sequential Monte Carlo method," in *IEEE International Conference of Engineering in Medicine and Biology Society*, Lyon, France, 22-26 Aug 2007, pp. 4185–4188.

[8] Q. Wen, J. Gao, and K. Luby-Phelps, "Multiple interacting subcellular structure tracking by sequential Monte Carlo method," in *IEEE International Conference on Bioinformatics and Medicine*, Silicon Valley, Nov 2007.

[9] Q. Wen, J. Gao, and K. Luby-Phelps, "Tracking multiple subcellular structures using a sequential Monte Carlo approach," *International Journal of Data Mining and Bioinformatics*, vol. 3, no. 3, 2009.

[10] Q. Wen, J. Gao, and K. Luby-Phelps, "Markov chain Monte Carlo data association for merge and split detection in tracking protein clusters," in *International Conference on Pattern Recognition (15% oral acceptance rate)*, vol. 1, Hong Kong, 2006, pp. 1030–1033.

[11] A. Dufour, V. Shinin, S. Tajbakhsh, N. Guillen-Aghion, J.-C. Olivo-Marin, and 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, Sept. 2005.

[12] A. P. Goobic, J. Tan, and S. T. Acton, "Image stabilization and registration for tracking cells in the microvasculature," *IEEE Transactions on Biomedical Engineering*, vol. 52, no. 2, pp. 287–299, Feb. 2005.

[13] N. Ray, S. T. Acton, and K. Ley, "Tracking leukocytes in vivo with shape and size constrained active contours," *IEEE Transactions on Medical Imaging*, vol. 21, pp. 1222–1235, 2002.

[14] C. Zimmer, E. Labruyere, V. Meas-Yedid, N. Guillen, and J.-C. Olivo-Marin, "Segmentation and tracking of migrating cells in videomicroscopy withparamet-

ric active contours: a tool for cell-based drug testing," *IEEE Transactions on Medical Imaging*, vol. 21, no. 10, pp. 1212–1221, Oct. 2002.

[15] V. Awasthi, K. W. Doolittle, G. Parulkar, and J. G. McNally, "Cell tracking using a distributed algorithm for 3-D image segmentation," *Bioimaging*, vol. 2, no. 2, pp. 98–112, 1994.

[16] C. M. Anderson, G. N. Georgiou, I. E. G. Morrison, G. V. W. Stevenson, and R. J. Cherry, "Tracking of cell surface receptors by fluorescence digital imaging microscopy using a charge-coupled device camera," *Journal of Cell Science*, vol. 101, no. 2, pp. 415–425, 1992.

[17] U. Kubitscheck, O. Kckmann, T. Kues, and R. Peters, "Imaging and tracking of single GFP molecules in solution," *Biophysical Journal*, vol. 78, pp. 2170–2179, 2000.

[18] G. Rabut and J. Ellenberg, "Automatic real-time three-dimensional cell tracking by fluorescence microscopy," *Journal of Microscopy*, vol. 216, pp. 131–137, 2004.

[19] C. C. Reyes-Aldasoro, S. Akerman, and G. M. Tozer, "Measuring the velocity of fluorescently labelled red blood cells with a keyhole tracking algorithm," *Journal of Microscopy*, vol. 229, pp. 162–173, 2008.

[20] D. Sage, F. R.Neumann, F. Hediger, S. M. Gasser, and M. Unser, "Automatic tracking of individual fluorescence particles: Application to the study of chromosome dynamics," *IEEE Transactions on Image Processing*, vol. 14, pp. 1372–1383, Sep 2005.

[21] D. Thomann, J. Dorn, P. K. Sorger, and G. Danuser, "Automatic fluorescent tag localization ii: improvement in super-resolution by relative tracking," *Journal of Microscopy*, vol. 211, no. 3, pp. 230–248, Sep. 2003.

[22] M. K. Cheezum, W. F. Walker, and W. H. Guilford, "Quantitative comparison of algorithms for tracking single fluorescent particles," *Biophysical Journal*, vol. 81, pp. 2378–2388, October 2001.

[23] M. J. Saxton, "Single-particle tracking:applications to membrane dynamics," *Annual Review of Biophysics and Biomolecular Structure*, vol. 26, pp. 373–399, June 1997.

[24] F. Briquet-Laugier, C. Boulin, and J.-C. Olivo-Marin, "Analysis of moving biological objects in video microscopy sequences," in *Proc. IS&T/SPIE Conference on Capture, Analysis and Display of Image Sequences*, vol. 3642, San Jose, CA, Jan. 1999, pp. 4–11.

[25] R. T. Collins, Y. Liu, and M. Leordeanu, "Online selection of discriminative tracking features," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 10, pp. 1631–1643, 2005.

[26] C. Schmid, R. Mohr, and C. Bauckhage, "Evaluation of interest point detectors," *International Journal of Computer Vision*, vol. 37, no. 2, pp. 151 – 172, Jun 2000.

[27] B. Ristic, S. Arulampalam, and N. Gordon, *Beyond the Kalman Filter: Particle Filters for Tracking Applications*. Artech House, 2004.

[28] M. Arulampalam, S. Maskell, N. Gordon, and 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, Feb. 2002.

[29] A. Doucet, "On sequential simulation-based methods for Bayesian filtering," Cambridge University Department of Engineering, Tech. Rep. CUED/F-INFENG/TR.310, 1998.

[30] A. Doucet, J. D. Freitas, and N. Gordon, Eds., *Sequential Monte Carlo Methods in Practice*. New York: Springer, 2001.

[31] S. Haykin and N. de Freitas, "Special issue on sequential state estimation: From kalman filters to particle filters," in *Proceedings of the IEEE*. IEEE, Mar. 2004, vol. 92, no. 3, pp. 399–574.

[32] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," in *IEE Proceedings F Radar and Signal Processing*, vol. 140, no. 2, April 1993, pp. 107–113.

[33] J. Lichtenauer, M. Reinders, and E. Hendriks, "Influence of the observation likelihood function on particle filtering performance in tracking applications," in *IEEE International Conference on Automatic Face and Gesture Recognition*, 2004, pp. 767–772.

[34] M. Isard and A. Blake, "Condensation - conditional density propagation for visual tracking," *International Journal of Computer Vision*, vol. 29, no. 1, pp. 5–28, Feb. 1998.

[35] Y. Rui and Y. Chen, "Better proposal distributions: Object tracking using unscented particle filter," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, 2001, pp. 786–793.

[36] Y. Wu and T. Huang, "A co-inference approach to robust visual tracking," in *International Conference on Computer Vision*, vol. 2, July 7-14 2001, pp. 26–33.

[37] H. Fei, "A hybrid hmm/particle filter framework for non-rigid hand motion recognition," in *IEEE International Conference on Acoustics, Speech, and Signal Processing*, vol. 5, May 17-21 2004, pp. 889–892.

[38] K. Nummiaro, E. Koller-Meier, and L. V. Gool, "Object tracking with an adaptive color-based particle filter," *Lecture Notes in Computer Science*, vol. 2449/2002, pp. 353–360, 2002.

[39] P. Perez, J. Vermaak, and A. Blake, "Data fusion for visual tracking with particles," *Proceedings of the IEEE*, vol. 92, no. 3, pp. 495–513, Mar. 2004.

[40] S. Birchfield, "Elliptical head tracking using intensity gradients and color histograms," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, June 1998, pp. 232–237.

[41] L. Vincent, "Morphological grayscale reconstruction in image analysis: Applications and efficient algorithms," *IEEE Transaction on Image Processing*, vol. 2, no. 2, pp. 176–201, April 1993.

[42] Y. Liu, X. Zhou, and W. Ma, "Extracting texture features from arbitrary-shaped regions for image retrieval," in *IEEE International Conference on Multimedia and Expo*, vol. 3, June 27-30 2004, pp. 1891–1894.

[43] M. J. Swain and D. H. Ballard, "Color indexing," *Internaltional Journal of Computer Vision*, vol. 7, no. 1, pp. 11–32, 1991.

[44] N. Sebe, Q. Tian, E. Loupias, M. Lew, and T. Huang, "Color indexing using wavelet-based salient points," in *IEEE Workshop on Content-based Access of Image and Video Libraries*, June 12 2000, pp. 15–19.

[45] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proceedings of The Fourth Alvey Vision Conference*, Manchester, 1988, pp. 147–151.

[46] D. B. Ried, "An algorithm for tracking multiple targets," *IEEE Transactions on Automatic Control*, vol. AC-24, no. 6, pp. 843–854, Dec 1979.

[47] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe, "Sonar tracking of multiple targets using joint probabilistic data association," *IEEE Journal of Oceanic Engineering*, vol. OE-3, no. 3, pp. 173–184, July 1983.

[48] I. J. Cox and S. Hingorani, "An efficient implementation of Reid's multiple hypotheses tracking algorithm and its evaluation for the purpose of visual tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 18, no. 2, pp. 138–150, 1996.

[49] H. Gauvrit, J. P. L. Cadre, and C. Jauffret, "A formulation of multi-target tracking as an incomplete data problem," *IEEE Transactions on Aerospace and Electronic Systems*, vol. 33, no. 4, pp. 1242–1257, 1997.

[50] I. J. Cox, "A review of statistical data association techniques for motion correspondence," *International Journal of Computer Vision*, vol. 10, no. 1, pp. 53–66, 1993.

[51] K. Smith, D. Gatica-Perez, and J.-M. Odobez, "Using particles to track varying number of interacting people," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, June 20-25 2005, pp. 962–969.

[52] Z. Khan, T. Balch, and F. Dellaet, "MCMC-based particle filtering for tracking a variable number of interacting targets," *IEEE Transaction on Pattern Analysis and Machine Intelligence*, vol. 27, no. 11, pp. 1805–1819, Nov. 2005.

[53] T. Yu and Y. Wu, "Collaborative tracking of multiple targets," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 1, 27 June-2 July 2004, pp. I–834 – I–841.

[54] S. Z. Li, *Markov Random Field Modeling in Computer Vision.* Springer, 1995.

[55] H. Sidenbladh and S.-L. Wirkander, "Tracking random sets of vehicles in terrain," in *IEEE Workshop on Multi-Object Tracking*, Madison, WI, USA, 2003.

[56] J. Vermaak, S. J. Godsill, and A. Doucet, "Radial basis function regression using trans-dimensional sequential Monte Carlo," in *IEEE Workshop on Statistical Signal Processing*, 28 Sept.-1 Oct. 2003, pp. 545–548.

[57] P. J. Green, "Reversible jump Markov chain Monte Carlo computation and Bayesian model determination," *Biometrika*, vol. 82, no. 4, pp. 711–732, 1995.

[58] M. K. Pitt and N. Shephard, "Filtering via simulation: auxiliary particle filters," *Journal of the American Statistical Association*, vol. 94, no. 446, pp. 590–599, Jun. 1999.

[59] N. Gordon, D. Salmond, and A. Smith, "Novel approach to nonlinear/non-Gaussian Bayesian state estimation," *IEE Proc.-F*, vol. 140, no. 2, pp. 107–113, April 1993.

[60] W. R. Gilks, S. Richardson, and D. J. Speigelhalter, Eds., *Markov Chain Monte Carlo in Practice*. Chapman & Hall, 1996.

[61] R. C. Gonzalez and R. E. Woods, *Digital Image Processing*, 2nd ed. Prentice Hall, 2002.

[62] E. R. Dougherty and R. A. Lotufo, *Hands-on Morphological Image Processing*. SPIE Press, 2003.

[63] T. Zhao and R. Nevatia, "Tracking multiple humans in crowded environment," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, June 27-July 2 2004, pp. 406–413.

[64] S. Gopinath, Q. Wen, N. Thakoor, K. luby Phelps, and J. Gao, "A statistical approach for intensity loss compensation of confocal microscopy images," *Journal of Microscopy*, vol. 230, pp. 143–159, 2008.

[65] D. Comaniciu, M. Visvanathan Ramesh, and P. Meer, "Kernel-based object tracking," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 5, pp. 564–577, May 2003.

[66] J. J. Craig, *Introduction to Robotics Mechanics and Control*, 2nd ed. Addison-Wesley Publishing Company, 1989.

[67] T. Yang, S. Z. Li, Q. Pan, and J. Li, "Real-time multiple objects tracking with occlusion handling in dynamic scenes," in *Proceedings of Computer Vision and Pattern Recognition 2005*, vol. 1, no. 1, 20-25 June 2005, pp. 970 – 975.

[68] A. Genovesio and J.-C. Olivo-Marin, "Split and merge data association filter for dense multi-target tracking," in *International Conference on Pattern Recognition*, vol. 4, Aug. 23-26 2004, pp. 677–680.

[69] Z. Khan, T. Balch, and F. Dellaert, "Multitarget tracking with split and merged measurements," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 1, 20-25 June 2005, pp. 605 – 610.

[70] J. A. Withers and K. A. Robbins, "Tracking cell splits and merges," in *Proceedings of the IEEE Southwest Symposium on mage Analysis and Interpretation*, 8-9 April 1996, pp. 117 – 122.

[71] S. Oh, S. Russell, and S. Sastry, "Markov chain Monte Carlo data association for general multiple-target tracking problems," in *43rd IEEE Conference on Decision and Control (CDC)*, vol. 1, 14-17 Dec 2004, pp. 735–742.

[72] S. Cong, L. Hong, and D. Wicker, "Markov chain Monte Carlo approach for association probability evaluation," *IEE Proceedings-Control Theory and Applications*, vol. 151, no. 2, pp. 185–193, March 2004.

[73] G. Welch and G. Bishop, "An introduction to the kalman filter," 2004. [Online]. Available: www.cs.unc.edu/ welch/kalman/kalmanIntro.html

BIOGRAPHICAL STATEMENT

Quan Wen was born in Sichuan province, China, in 1972. He received his B.S. degree with highest honor in automatic control, in 1994, his M.S. in computer application, in 1997, both from Sichuan University, China. From 1997 to 2002, he was with the New Technology Development Center (NTDC) of China Telecom, Shenzhen, focusing on the killer applications of internet such as Video On Demand (VOD) and WebTV. He won the NTDC's Outstanding Employee Award in 2001. During the whole year of 1998, He also worked for Integrated Network Corporation (INC), New Jersey, on broadband VOD systems using Very High Speed Digital Subscriber Line (VDSL) technology. He joined the Ph.D. program of the Department of Computer Science and Engineering at the University of Texas at Arlington in Spring, 2003. He is a student member of IEEE. His research interests are mathematical modeling of static and dynamic high dimensional noisy data, and its application in biomedical image and video processing. He also has great interest in the content-based image and video retrieval.