UNIVERSITY OF CALIFORNIA

Santa Barbara

Automated Microtubule Tracking and Analysis

A dissertation submitted in partial satisfaction of the
requirements for the degree of Doctor of Philosophy
in Electrical and Computer Engineering

by

Motaz El-Saban

Committee in charge:

Professor B. S. Manjunath, Chair

Professor Shivkumar Chandrasekaran

Professor Kenneth Rose

Professor Stuart Feinstein

Professor Leslie Wilson

March 2006

The dissertation of Motaz El-Saban is approved.

Professor Shivkumar Chandrasekaran

Professor Kenneth Rose

Professor Stuart Feinstein

Professor Leslie Wilson

Professor B. S. Manjunath, Committee Chair

March 2006

Automated Microtubule Tracking and Analysis

Copyright © 2006

by

Motaz El-Saban

To my parents, wife and to the light of my life Noor.

ACKNOWLEDGMENTS

I take this opportunity to thank my advisor, professor B. S. Manjunath, for his full support during the course of my graduate studies at UCSB. His encouraging words have been always guiding and motivating me to finish this work. I would like to express my thanks for my committee members for their constructive comments and help throughout the research work. I am very grateful to Professor Stuart Feinstein and Leslie Wilson for the time and effort they have put in explaining to me the biological basis for the work presented in this thesis. With their support and devotion, this work was made possible. The comments and suggestions of Professor Shiv Chandrasekaran and Kenneth Rose were very valuable in completing this work. Beside the committee members, I would like also to thank Dr. Brian Matsumoto, the director of the Integrated Microscopy Facility at UCSB, for serving as an instructor for an excellent course on microscopy techniques which clarified to me many details on the image acquisition level. Without the good environment provided by my parents, wife and little daughter I could not imagine reaching the point of graduation. I also would like to thank my lab colleagues both at the vision research lab and at the center of bio-image informatics. On top of the list comes Alphan Altinok, Austin Peck and Dr. Charles Kenney for lengthy discussions and collaborative research work, much of which went into my dissertation. I express my thanks to Baris Sumengen for his support and help with ideas and directions. Other students and researchers who made my graduate life at UCSB an enjoyable experience include: Dr. Zhiqiang Bi, Dr. Peter Markiewicz, Luca Bertelli,

Marco Zuliani, Sitaram Bhagavaty, Xinding Sun, Dimitry Fedorov, Jiyun Biyun, Tapas Manna, Dr. Kristian Kvilekval, Aleem Siddiqui, Arnab Battacharya, Vebjorn Ljosa, Nhat Vu, Ibrahim Khalil, Shawn Newsam and Wei Niu who shares part of the work on temporal rule mining presented in this thesis.

VITA

MOTAZ EL-SABAN

March 2006

EDUCATION

2006	*Doctor of Philosophy* Department of Electrical and Computer Engineering University of California, Santa Barbara, California
2004	*Master of Science* Department of Electrical and Computer Engineering University of California, Santa Barbara, California
2000	*Master of Science* Department of Electronics and Communication Engineering Cairo University, Faulty of Engineering Giza, Egypt
1997	*Bachelor of Science* Department of Electronics and Communication Engineering Cairo University, Faulty of Engineering Giza, Egypt

PUBLICATIONS

M. E-Saban et al. “Automated tracking and modelling of microtubule dynamics,” IEEE International Symposium on Biomedical Imaging (ISBI 2006).

B. S. Manjunath, B. Sumengen, Z. Bi, J. Byun, M. El-Saban, D. Fedorov, N. Vu, “Towards Automated Bioimage Analysis: from features to semantics,” IEEE International Symposium on Biomedical Imaging (ISBI 2006), invited paper.

A. Altinok, M. E-Saban et al. “Activity Recognition in Microtubule Videos by Mixture of Hidden Markov Models,” IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006).

S. Bhagavaty and M. A. El Saban. “SketchIt: Basketball Video Retrieval Using Ball Motion Similarity,” IEEE Pacific-Rim Conference on Multimedia (PCM 2004).

M. A. El Saban and B. S. Manjunath. “Interactive Segmentation Using Curve Evolution And Relevance Feedback,” IEEE Signal Processing Society International Conference on Image Processing (ICIP 2004).

M. A. El Saban and B. S. Manjunath. “Video Region Segmentation by Spatio-temporal watersheds,” IEEE Signal Processing Society International Conference on Image Processing (ICIP 2003).

M. A. El Saban, S. Abd El-Azeem and M. Rashwan. “A new video coding scheme based on the H.263 standard and entropy constrained vector quantization,” Cairo University, Faculty of engineering journal, Nov. 2000.

M. A. El Saban, S. Abd El-Azeem and M. Rashwan. “A new video coding scheme based on the H.263 standard and entropy constrained vector quantization,” ICII 2000, Kuwait, Nov. 2000.

Abstract

Automated Microtubule Tracking and Analysis

by

Motaz El-Saban

Microtubules are major components of the cytoskeleton and play an important role in a number of cellular functions such as maintaining cell shape, cell division and transport of various molecules. Abnormal dynamic behavior of microtubules has been associated with neuro-degenerative diseases (e.g., Alzheimer) and cancer. Researchers study the dynamics of microtubules under different experimental conditions including different drug treatments, and using time sequence images from fluorescence microscopy. At present the dynamics of microtubules are quantified using simple first and second-order statistical measures of the length variations of manually tracked microtubules. The current analysis being mostly done manually, is quite laborious and time-consuming. Besides, the number of microtubules that one can track with manual methods is limited. In the first part of the thesis, we propose novel tools for automated detection and tracking of microtubules. A multiframe graph-based approach is proposed to tackle tracking issues, and our results demonstrate the robustness of the proposed approach to occlusions and intersections.

In the second part of the thesis, we propose the use of statistical modeling tools for a better understanding of the underlying molecular mechanisms of microtubule dynamics. Prototype models are estimated for various experimental conditions by training

hidden Markov models (HMMs) on the microtubule tracking data. Furthermore, these models are used to quantify similarities between experimental conditions. Additionally, temporal association rules are derived to characterize frequent patterns in the microtubule dynamics under different experimental conditions. The extraction of frequent patterns leads to a better understanding of how an experimental condition, such as the application of a drug, modulates microtubule dynamics.

Contents

List of Figures

List of Tables

Chapter 1

Introduction and Motivation

Recent advances in cell biology research and the emergence of new microscopy techniques that are able to image cells at the molecular level have resulted in large collections of digital bio-molecular images and videos. The vast majority of the image analysis, at present, is done manually. However, in order to fully harvest the information present in the growing number of image/video collections –both qualitatively and quantitatively – much progress is needed on the automated analysis of the data. Our primary goal in this thesis is to develop new computer vision and information extraction tools that can facilitate the analysis of bio-molecular image sequences. While the specific datasets that we study here are concerned with subcellular structures known as microtubules and their dynamics, the basic tools that we develop are applicable to a broader class of such images.

1.1 Microtubule Data

Microtubules (MTs) are cylindrical, protein polymers found in essentially all eukaryotic cells. Microtubules are one of the components of the cytoskeleton – the cell skeleton. Microtubules are composed of two subunits of monomers, the α and β tubulin protein. Microtubules are highly dynamic polymers in the sense that their ends can be growing (polymerizing) or shortening (depolymerizing) depending on the needs of the cell. Polymerization (depolymerization) is the process by which new polymer units are added (subtracted) from an existing microtubule at one of its ends. This result in either growing or shortening of the microtubule.

Investigators generally visualize microtubule dynamics in cells using time-lapse fluorescence microscopy of cells that have been engineered to possess fluorescent tubulin. Time-lapse fluorescence is a microscopy technique used for imaging live cells. Image frames are acquired at regular time intervals by illuminating the biological sample with a beam of light. As a result, the tubulin structures in the cells emit back, by fluorescence, another light wave that is imaged by the microscope. The image sequences used in this thesis are taken at 4 second intervals, with about 30-60 frames per stack of images. Fig. 1.1 shows two sample frames in such a sequence. The image frames have dimensions of 512 $\times$ 600 pixels with one pixel corresponding to a 100 nM $\times$ 100 nM area (approximately) and an intensity resolution of 12 bits/pixel.

The observations that are of interest here include the growth and shortening of these tiny hair-like structures over a period of time for a specific experimental condition such

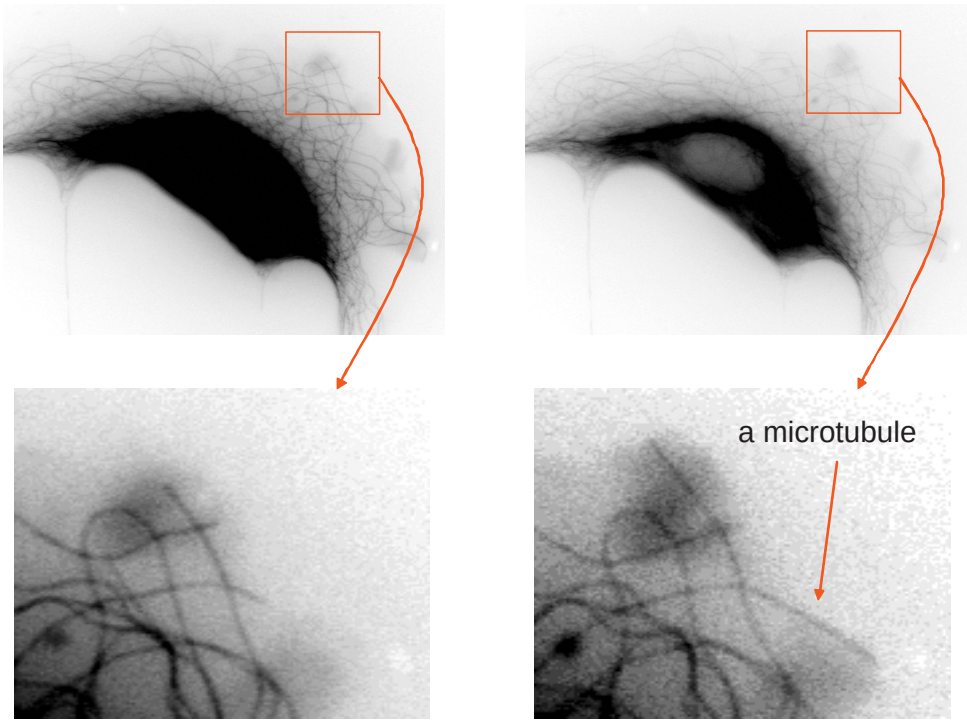


Figure 1.1: Top: frame 4 (right) and 24 (left) from a microtubule stack. Bottom: zoom-in on the same area in both frames. The small windows show the high level of noise, contrast and illumination changes between the two frames and the significant “growth” of a microtubule.

as the administration of a drug (See Fig. 1.1). The current approach used to quantify the growth and shortening of microtubules is to record the length of microtubules in every frame of a stack. The microtubule length is approximated by the Euclidean distance between the microtubule tip and a fixed reference point. The position of the microtubule tip is manually tracked from frame to frame. From the length computation in the video frames, simple statistical averages of dynamic parameters, such as growth and shortening rates, are computed. More details on the imaging of the microtubules and their dynamics are discussed in Chapter 2.

1.2 Data Analysis Issues

The extraction of dynamic parameters from microtubule videos under a given experimental condition consists of manual tip tracking and the computation of first order statistical measures. There are several issues in both the manual tracking and dynamic parameter computation:

- The manual tracking is laborious and time consuming. Hence, usually a relatively small number of microtubules are tracked per stack. This leaves open the question of whether or not current sampling provides a sufficient number of representative microtubules for drawing reliable conclusions.

- Manual tracking can be subjective and introduces inter-observer variations. Two individuals asked to track the same microtubule can mark different tip positions

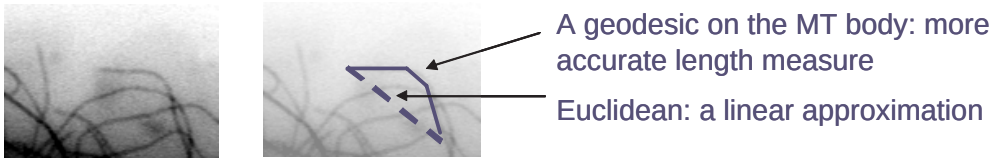
depending on their accuracy and the image noise level.

- The length of a microtubule is currently approximated by computing the Euclidean distance between the tip location and an arbitrarily selected reference point on the microtubule body. This length computation is inaccurate since the microtubule shape is not always a straight line because microtubules can bend or move laterally from frame to frame (See Fig. 1.2). A more accurate method would compute the length as the shortest path along the microtubule body (i.e. a *geodesic*).

- It is difficult to test hypotheses on large volumes of stacks such as analyzing the effect of different test conditions on microtubule shape, or quantifying the effect of neighboring microtubules as only the microtubule ends are currently recorded.

- The simple first order statistical parameters computed may not uncover critical factors affecting microtubule dynamics. Using advanced statistical tools and pattern mining techniques, one may discover fundamentally important but less obvious information.

Based on the above issues, we will discuss next the major challenges that face the areas of potential benefit for the application of intelligent information processing techniques for microtubule dynamic studies.



(a) (b)

Figure 1.2: Estimation of the length of a microtubule is currently based on the Euclidean distance between the tip and a reference point. The original frame shown in (a), while (b) shows the inaccuracy of approximating the microtubule length using a Euclidean distance. A better length estimate computes the distance as the shortest path along the microtubule body (i.e. a geodesic).

1.3 Information Processing Tasks

1.3.1 Enhancement and Segmentation of Microtubule Frames

Given the low signal-to-noise ratio, nonuniform illumination, photo-bleaching and out-of-focus problems typical to time-lapse fluorescent microtubule videos, a crucial step in automated processing of these frames is to enhance their quality for further processing. Additionally, researchers generally track microtubules near the cell periphery so as to maximize the resolution of each microtubule. Hence, in order to automate the process of detection and tracking of microtubules, we need to first extract the cell region that contains peripheral microtubules through image/video segmentation.

1.3.2 Automating Microtubule Tracking

Automating the detection and tracking of microtubules in videos is an obvious first task. With an automated approach, one should be able to track more microtubules in a given video than manually feasible. Further, it removes the potential subjective bias in the manual tracking process. As we noted before, microtubules appear as tubular structures in the image frames. The shape of microtubules varies widely within the same cell. An accurate estimation of the length of the microtubule should take into account its curvilinear structure. The currently widely used manual tip tracking and computing the associated statistics ignores this effect. In addition, microtubules undergo large changes in length from frame to frame because of growth or shortening at either microtubule end. These large changes in length, together with frequent occlusions, pose significant problems to any tracking method. Further, the images of microtubules have a low signal-to-noise ratio and exhibit nonuniform illumination both spatially and temporally, adding to the difficulty in their detection and tracking.

1.3.3 Detecting Non-obvious Patterns and Similarities

By developing automated tracking tools for microtubule videos, a large volume of tracking data will become available for further processing. Questions related to common behavior between different experimental conditions can be answered with a high degree of statistical significance due to the availability of a large volume of data. For example, construction of models for microtubule behavior may be very useful for pre-

dicting microtubule behavior under test conditions that have not been experimentally assessed before. Additionally, by applying data mining tools on the tracking data, temporal and spatial patterns of dynamics can be extracted for different conditions. As an example, patterns of microtubule dynamics can be used to understand how a drug of interest affects microtubule behavior.

1.4 Summary of Contributions

We present in this thesis a set of new computational tools that target the above information processing challenges. Specifically we made contributions in the following areas of automated analysis of microtubule dynamics:

1. Extraction of the cell layer containing microtubules of interest.
2. Automated microtubule detection in a single frame.
3. Fully automated tracking of microtubules.
4. Modeling and pattern discovery in microtubule videos using statistical tools.

1.4.1 Microtubule Layer Detection

Since the microtubule polymer mass near the nucleus is usually very high, resulting in high fluorescence strength, one cannot resolve individual microtubules in the center region of the cell. Hence, we are usually interested in microtubules growing or shortening near the cell membrane. We present an automated technique to extract a band

around the cell membrane -the microtubule layer- through temporal clustering in the frequency domain. We consider the microtubule video of T frames as a spatiotemporal volume. We construct frequency vectors at each pixel location (x, y) on the first frame. The frequency vectors are clustered into V clusters corresponding to regions of varying activity in the cell. Finally, we extract the clusters corresponding to the region of highest activity, which we call the microtubule layer. The microtubule layer extraction is detailed in Chapter 3.

1.4.2 Microtubule Detection

After extracting the microtubule layer, we present an automated approach to detect microtubules in this layer. The basic assumption about microtubules that enables extracting a binary mask of microtubule locations is that microtubules look as black curvilinear structures on a light background in an ideal scenario. We present a filtering-based approach to detect the microtubule polymer mass. A second derivative of Gaussian kernel matched to image locations at different orientations should reveal the microtubule structure while reducing background noise. Based on the filtering output, we extract a microtubule polymer mass binary mask, from which we devise an approach to detect microtubule tips that are used in subsequent tracking. The details of microtubule detection are provided in Chapter 3.

1.4.3 Fully Automated Tracking of Microtubules

Given the limitations of the current manual approach to track microtubules in videos, we propose a fully automated, robust tracking technique in Chapter 4. The technique performs well under low signal to noise ratio, varying illumination conditions and can handle occlusions and intersections. By performing a simultaneous tracking for all the microtubules and for the full time duration of the video, we are able to resolve tracking conflicts and to overcome microtubule tip occlusion and mis-detection. This global approach is formulated in a graph-based context.

1.4.4 Modeling and Pattern Discovery

The above contributions deal with the automation of the tracking step while not attempting to provide new analysis tools to microtubule studies. Using the tracking data, we propose new tools that can further our understandings of the underlying cell regulatory mechanisms for microtubule dynamics. Towards this end, we propose the use of of hidden Markov models (HMMs) as a powerful statistical modeling tool to describe microtubule tracking observations. By modeling microtubule behavior using HMMs, we are equipped with a measure of similarity between effects of different test subjects - such as regulatory microtubule associated proteins (MAPs) and drugs - on microtubule dynamics. Furthermore, we propose a temporal rule mining algorithm that can detect common pattern of dynamics for different experimental conditions, thus providing previously unattainable valuable information to biologists. The proposed modeling and

pattern discovery tools for microtubule videos are detailed in Chapter 5.

1.5 Dissertation Outline

The thesis is organized as follows. Chapter 2 presents details on the current techniques used to image and analyze microtubule dynamics, and some recent examples of microtubule studies. The steps used for microtubule sample preparation are also discussed. Chapter 3 reviews the related work on automated microtubule detection and tracking. It gives an overview on relevant related work on object tracking using active contours and curvilinear structure tracking. In addition, Chapter 3 presents our new method for extracting tip features from microtubule videos. Chapter 4 gives a detailed description of the new tracking approach based on spatio-temporal approach for all microtubules. The tracking is formulated using a graph context augmented with an active contour to handle lateral microtubule motion. Qualitative and quantitative experimental results are presented for microtubule tracking. Chapter 5 details the proposed methods for temporal pattern extraction and modeling of microtubules, thus introducing new analysis tools not previously available. The models and rules constructed agree with biological findings and provide novel insights into cell mechanisms for regulating microtubule dynamics. Furthermore, the constructed models of microtubule dynamics can be used to quantify commonalities between different experimental conditions. Finally, summary and future research directions are outlined in Chapter 6.

Chapter 2

Imaging and Studying Microtubules

A detailed review of current research on microtubule dynamics is presented in this chapter. The chapter begins with a detailed description of the structure and function of microtubules and how they are important for cell biology and the cell cycle. We then discuss three recent representative research studies on microtubule dynamics to provide the reader with a perspective on the current state-of-the art. Next, we present the two stages involved in obtaining microtubule videos, those being: a) sample preparation and b) microscopy imaging of dynamics. Finally, we list the main parameters describing microtubule dynamics.

2.1 Structure and Function of Microtubules

Microtubules (MTs) are protein polymers corresponding to one of three major components of the cytoskeleton – the cell skeleton. In terms of structure, microtubules are

hollow cylinders of 15 nm and 25 nm internal and external diameters respectively. The length of the microtubules can extend up to 1000 times its diameter. The hollow tube is composed of α tubulin - β tubulin dimer subunits. A schematic of a microtubule is shown in Fig. 2.1.

The α and β tubulins exist as different isotypes at the protein level such as αI, αII, βI and βIII. The different isotypes vary in terms of their structure. One full turn of a microtubule is made out of 13 individual tubulin molecules. The microtubule has two ends distinctly noted as the plus and minus ends (See Fig. 2.1). The distinction between the two ends gives the rise to the polarized structure to the microtubules. Microtubules are constantly increasing and decreasing in length through addition and loss of subunits at their ends, a phenomenon called dynamic instability. A particular case occurs when the plus end is polymerizing at the same rate that the minus end is depolymerizing, which is called treadmilling. This phenomenon gives the appearance of a net motion of the microtubule.

In cell biology, an important concept is that structure determines function. Microtubules are no exception. Tubular in structure and highly dynamic, microtubules are essential for numerous critical cellular functions, including various intracellular transport processes, cell movement and cell shape changes. For example, motor proteins, bind to microtubules and "read" its polarity and move from one end to the other, thereby delivering cargo to specific cellular domains. Especially notable among the many functions of microtubules is the essential role of microtubules in separating chromosomes to

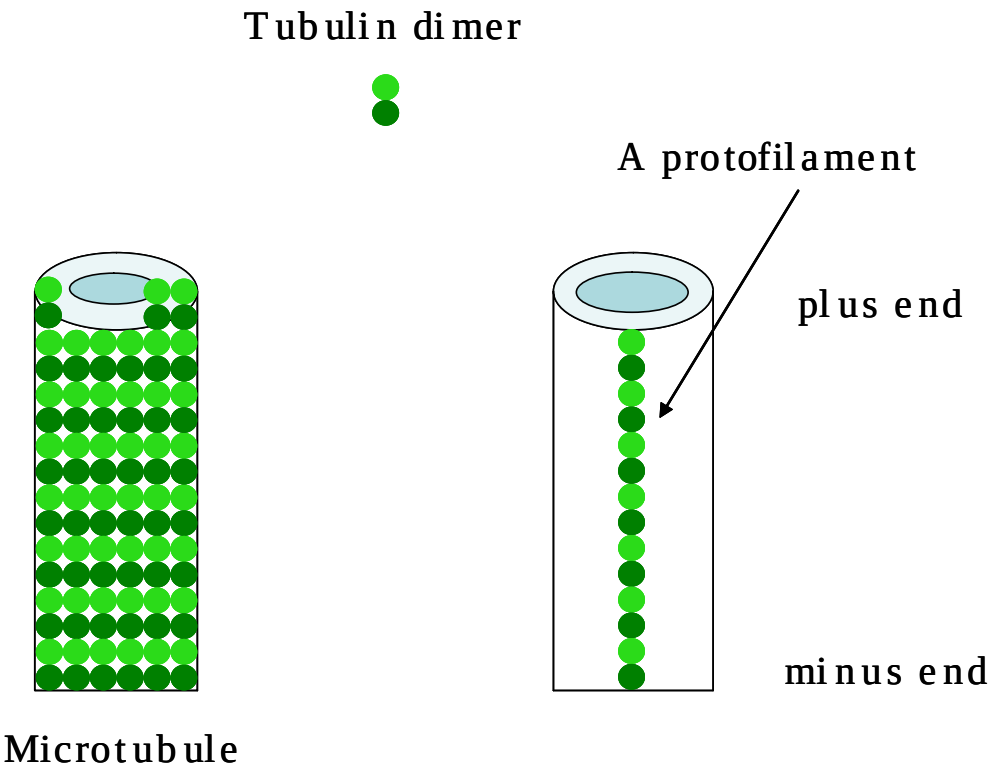


Figure 2.1: A schematic of an example microtubule showing the cylindrical structure and the constituting dimers (α and β tubulin shown as light and dark green spheres respectively (after [8]).

the two daughter cells during cell division. Recent work has demonstrated convincingly that proper regulation of the growing and shortening dynamics of microtubules is essential in order to achieve proper chromosome segregation (reviewed in [62]). Indeed, the highly effective anti-cancer drug paclitaxel works because it suppresses microtubule dynamics sufficiently to interfere with normal chromosome segregation. Recent work has also suggested that mis-regulation of microtubule dynamics may underlie Alzheimer's disease and related dementias (reviewed in [43]). Thus, the regulation of microtubule dynamics is a key element in both normal and pathological cell biology.

2.2 Research on Microtubule Dynamics

Owing to their many important roles, many research groups study microtubules in terms of structure, function and dynamic behavior under different experimental conditions [32], [38], [65], [97], [107]. Different agents can affect the dynamic behavior of microtubules. Example of such agents include microtubule associated proteins (MAPs) such as tau, MAP1A and MAP1B and drugs such as paclitaxel, colchicine, epothilone B and vinblastine. We used microtubule videos from [65] and [38] for the results reported in this thesis in Chapter 4 and Chapter 5.

2.2.1 Effect of Tau on Microtubule Dynamics

One class of microtubule associated proteins (MAPs) binds to microtubules and regulates their dynamic behavior. For example, the neural microtubule-associated protein

tau binds and stabilizes microtubules. Tau exists in two isoforms, known as 3-repeat and 4-repeat tau. While only 3-repeat tau is expressed in fetal brains, both 3- and 4-repeat tau are expressed in adult human brain at approximately equal levels. When the ratio between 3-repeat and 4-repeat tau is altered as a result of genetic mutation, neurodegenerative diseases occur such as FTDP-17. In the study by Bunker et al. [13], the authors investigated how both isoforms of tau regulate microtubule dynamics. It is observed that though both tau isoforms regulate dynamics, there are important differences. For example, 3-repeat tau has very little effect on the shortening rate of microtubules, whereas 4-repeat tau potently suppressed it.

2.2.2 Suppression of Microtubule Dynamics using Epothilone B

One important focus of microtubule dynamic studies is to discover potential novel anti-cancer drugs by virtue of their ability to block cell division. Towards this end, the study in [65] targets the Epothilone B compound. The dynamic behavior of microtubule is quantified in terms of the percentage of cells that are blocked from continuing through mitosis – one of key the stages in cell division. The main idea is that if a cell is blocked at mitosis, then tumor cell proliferation is blocked. Epothilone B is observed to suppress microtubule dynamics in a concentration dependent manner. At a concentration of 0.2 nM (nano molar), there is no mitotic arrest, whereas at 3.5 nM, there is half maximal arrest.

2.2.3 Effect of β-III on Resistance to Paclitaxel

It has been experimentally determined that paclitaxel suppresses microtubule dynamics. This is the main reason that paclitaxel is a very effective anti-cancer drug. However, it has been observed that after a period of treatment with paclitaxel, the cells start acquiring resistance against the suppressing effect of the drug on microtubule dynamics. It has been hypothesized that cell resistance is correlated with an over-expression of a β-tubulin isotype, known as βIII. In the study by Kamath et al. [38], the authors tested this hypothesis experimentally. The control condition was that of normal cells having βI-tubulin as the predominant β-isotype, comprising approximately 70$\%$ of the total β-tubulin pool. Paclitaxel effectively suppressed microtubule dynamics in the controls cells. However, by over-expressing βIII, paclitaxel had a very reduced effect on microtubule dynamics. The study concluded that, indeed, βIII is associated with acquired cell resistance to paclitaxel. Some of the videos used in this thesis comes from this β-III study [38].

It is worth noting here that it is very difficult to examine individual microtubule behavior during mitosis due to the bundling of microtubules that participate in the process. Hence, as an approximation, biologists often study microtubules during the interphase part of the cell cycle. The assumption is that microtubule behavior, under different test conditions, will be very close to the case of dividing cells.

2.3 Sample Preparation

Live cell fluorescence microscopy using GFP-tubulin is the technique used to image microtubules studied in this thesis. However other possibilities such as speckle, confocal, and differential interference contrast microscopy do exist and will be discussed later. We will briefly highlight the main components in microtubule sample preparation task:

1. **Type of the study:** *in vivo* and *in vitro*:

- *in vivo*: In "in vivo" studies, the microtubules are imaged in their cell environment, thus a labelling technique is required to visualize only the microtubules and not other cell components. The main advantage of *in vivo* studies is that microtubules are analyzed in their native environment, thus one can draw conclusions corresponding to the situation in living cells. On the other hand, *in vivo* studies of microtubules are affected by all the regulatory effects present in a living cell, thus making it very hard to control all the possible variables in the experimental paradigm.

- *in vitro*: In "in vitro" studies, the tubulin -from which microtubules are assembled- is purified away from other cellular components and thus microtubules are imaged separately. Usually techniques such as differential interference contrast (DIC) are used to image dynamics, which has a relatively low signal-to-noise ratio. The main advantage of *in vitro* studies is that microtubules are analyzed

separately from all other cellular factors, allowing more precise cause-and-effect relationships to be tested directly. On the other hand, the main disadvantage of *in vitro* studies is that they may or may not accurately mimick the real cellular situation.

Thus, one important strategy in biological research on microtubules is to study a test subject effect on microtubule dynamics first *in vitro* for making direct inferences. If an interesting observation is found, then *in vivo* study of the same test subject is carried out to assess the *in vitro* observation in live cells.

2. **Experimental sample:** *in vivo* studies can be performed in cultured cells, tissue samples or whole animals. Examples of cultured cells include breast cancer cells (MCF7) [51], cervical cancer cells (Hela) [52], and the rat adrenal cell line (PC12) [94].

3. **Test article/subject:** This is the main subject of a microtubule dynamics study. An example is studying the capability of a certain drug such as Epothilone B, in a concentration dependent manner, to block cells from progressing into mitosis, thus preventing cell division [65].

4. **Route of administration:** This is the technique by which the test subject is administrated to the experimental sample. For introduction into cells, the main techniques of administration are adding membrane permeable agents to bathing media, microinjection, and transfection (foreign DNA).

5. **Microtubule labelling:** This is the technique used to visualize the microtubules from the other cell components, so that only the microtubules are imaged. Techniques

of labelling include coupling of Green Fluorescent Protein (GFP) or Rhodamine to tubulin. The basic idea to genetically or chemically modify the tubulin protein so that it is fluorescent and thus emits light that can be imaged when excited with a light source of the appropriate wavelength.

2.4 Imaging Microtubule Dynamics

Though the main technique used in imaging the microtubule videos used in this thesis is fluorescence imaging, a few other alternative imaging techniques are discussed in the following.

2.4.1 Fluorescence Microscopy

In fluorescence microscopy, a sample of cells containing microtubules is labelled with a fluorescent protein such as GFP-tubulin. When a light beam hits the sample, the fluorescent molecules absorb photons. Their electrons are excited to higher orbits, remaining there for a short while, then returning back to the low level orbit and releasing a photon. The emitted photons from the sample are collected by a charge coupled device (CCD) array, and an image is formed (See, for example, Fig. 1.1). Hence fluorochrome materials act as energy transducers having both an absorption and an emission spectrum. One problem that can arise in live cell fluorescence imaging is referred to as photo-bleaching. Photo-bleaching refers to the fact that as the sample is hit by light and emits back photons, it loses the capability of fluorescing over time, and thus

becomes unimagable. Researchers attempt to compensate for photo-bleaching by minimizing and varying the amount of incident light on the sample, which poses significant challenges to any automated image processing technique to handle abrupt illumination changes.

2.4.2 Confocal Microscopy

The spatial resolution of regular fluorescence microscopy is limited by the optical efficiency in concentrating the light hitting the sample. In confocal fluorescence microscopy, a more advanced optical system which uses a more focussed laser beam hits point locations on the sample, thus providing higher spatial resolution and reducing out-of-focus illumination (See an example of microtubules in Fig. 2.2.) The drawback of this point scanning microscopy is that a raster scanning of the sample is required to image the full sample, thus limiting its application for high-speed capture of living cells used in dynamic studies.

2.4.3 Fluorescence Speckle Microscopy

Fluorescence Speckle Microscopy (FSM) is a recent technique that was developed to analyze the assembley/disassembly of macromolecular structures, such as actin filaments, and microtubules [27]. Random speckle appearance is obtained by generating very low densities of fluorescent labelling of the molecular structures. The main advantages of speckle microscopy are its ability to detect growth and shortening events

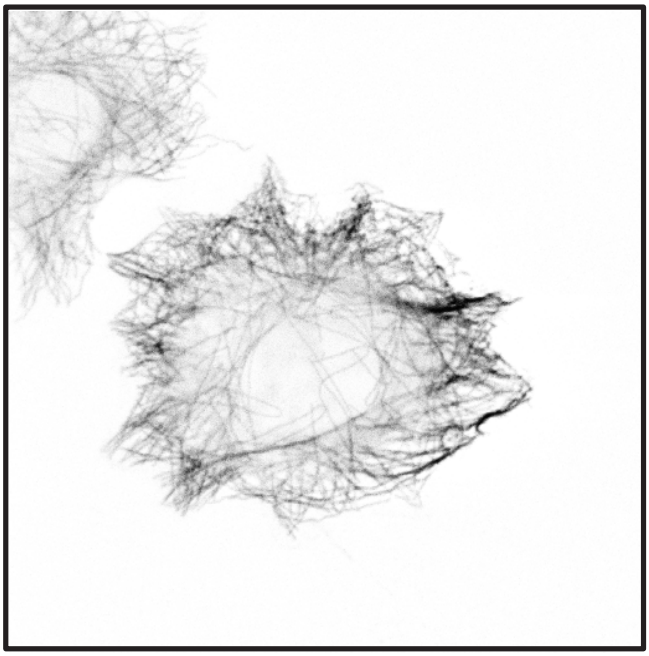


Figure 2.2: An example confocal image of microtubules (courtesy of [4]).

as well as translocations/sliding events, by only labelling sparse and randomly spaced points along the structure, and being able to track the speckles over time. Coupled with automated methods for tracking the speckles, this microscopy technique has proven to be very efficient in tracking actin network movement [40]. FSM also enhances the contrast of samples by reducing out-of-focus fluorescence. Initially, FSM was designed for conventional fluorescence microscopy and then adapted for confocal microscopy for better speckle contrast. However, one limitation of the FSM is that one can not track the microtubule tips which is extremely important in many microtubule dynamic studies.

2.4.4 Nomarski Differential Interference Contrast Microscopy (DIC)

The key to differential interference contrast microscopy is to use two polarizing filters and two prisms to actually obtain two images of the sample, offset by a small fraction of the wavelength of light. Having two images enables light comparison between them, which makes the Nomarski DIC capable of generating the highest resolution of a light microscope (See Fig. 2.4 for an example image). However, the main disadvantage of DIC is its low signal-to-noise image ratio, thus it is mostly used for *in vitro* studies where we image purified microtubules without any other interfering cell components. DIC is not suitable for *in vivo* studies, as the resulting low signal-to-noise ratio images would be very cluttered and it would not be possible to resolve individual microtubules.

2.4.5 Atomic Force Microscopy (AFM)

The previously discussed light microscopy techniques cannot visualize the individual molecular subunits of microtubules. However, visualization at this level can address many critical issues. A possible solution is to use a higher resolution imaging technique called atomic force microscopy (AFM), which can generate such as the ones in Fig. 2.3. In AFM, a very small tip is used to physically scan the surface of the microtubule. The force exerted on the tip is translated into a height measurement of the sample. Unfortunately, the time currently required to acquire one AFM image does not allow high speed imaging.

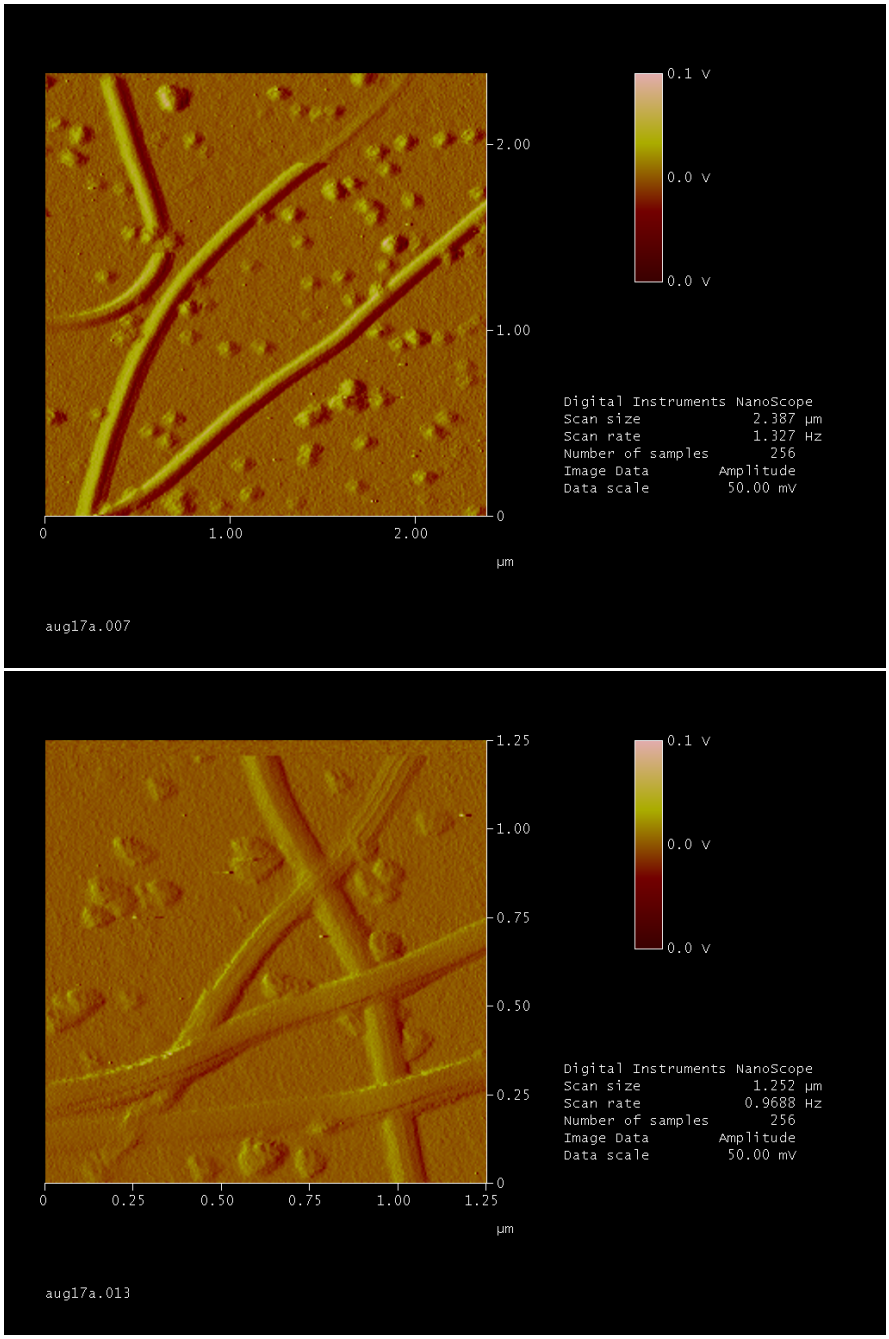


Figure 2.3: An example high-resolution AFM image of a number of microtubules (courtesy of [4]).

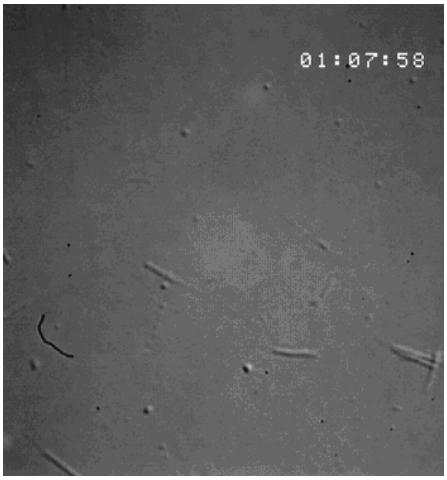


Figure 2.4: An example Nomarski Differential Interference contrast Microscopy image of microtubules (courtesy of [4]).

2.5 Analyzing Microtubule Dynamics

A typical microtubule dynamic study would generate several stacks that are then analyzed to quantify the growth and shortening events. The analysis step is mostly done manually, by tracking individual microtubule tips, one at a time, over the entire video. The length of a microtubule at any given frame is approximated by the Euclidean distance between the tracked tip and a reference point. From the length at each frame, a *life history plot* is generated. The life history plot is manually parsed into growth, shortening and attenuation events as shown in Fig. 2.5. A $GrowthEvent$ refers to periods of time where the microtubule is increasing in length. A $ShorteningEvent$ refers for periods of decrease in length. Attenuation refers to periods of neither significant growth nor shortening. *Rescue* is a transition from shortening to attenuation or growth. *Catastrophe* is transition from growth or attenuation to shortening. Table 2.1

summarizes some of the important parameters in a microtubule study.

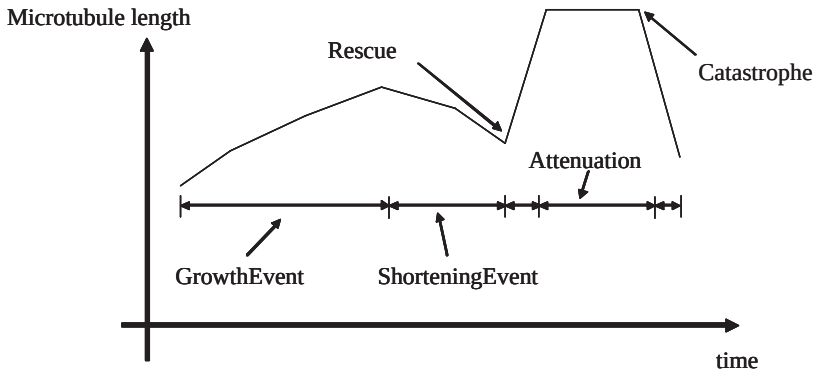


Figure 2.5: An example life history plot of a microtubule showing different dynamic events.

The mean and standard deviation (μ, σ) of all the shortening events belonging to one experimental condition are defined as:

$$\mu_{ShorteningRate} = \frac{1}{N_{SE}} \sum_{i=1}^{N_{SE}} ShorteningEventRate(i) \tag{2.1}$$

$$\sigma_{ShorteningRate} = \frac{1}{N_{SE}} \sum_{i=1}^{N_{SE}} (ShorteningEventRate(i) - \mu_{ShorteningRate})^2 \tag{2.2}$$

where $ShorteningEventRate = \frac{Total\ length\ decreased\ in\ \Delta t}{\Delta t}$

The mean and standard deviation (μ, σ) of all the growth events belonging to one experimental condition are defined as:

$$\mu_{GrowthRate} = \frac{1}{N_{GE}} \sum_{i=1}^{N_{GE}} GrowthEventRate(i) \tag{2.3}$$

$$\sigma_{GrowthRate} = \frac{1}{N_{GE}} \sum_{i=1}^{N_{GE}} (GrowthEventRate(i) - \mu_{GrowthRate})^2 \tag{2.4}$$

Parameter	Meaning
N_{MT}	Total number of MTs tracked
$GrowthEvent$	Time interval $[t, t + \Delta t]$ where a MT is growing in length.
$ShorteningEvent$	Time interval $[t, t + \Delta t]$ where a MT is decreasing in length.
$GrowthEventRate$ (of a $GrowthEvent$)	$\frac{Total\ Length\ grown\ in\ \Delta t}{\Delta t}$
$ShorteningEventRate$ (of a $ShorteningEvent$)	$\frac{Total\ length\ decreased\ in\ \Delta t}{\Delta t}$
T	Total MT track duration.
L_I (of a MT)	Total Length increase in T.
L_D (of a MT)	Total Length decrease in T.
$MTDynamicity$	$\frac{L_I + L_T}{T}$ for a MT.
N_{GE}	Number of $GrowthEvent$ for all MTs.
N_{SE}	Number of $ShorteningEvent$ for all MTs.
$ShorteningRate$	Shortening rate for all the MTs.
$GrowthRate$	Growth rate for all the MTs.
$Dynamicity$	Dynamicity for all the MTs.

Table 2.1: Microtubule (MT) dynamic parameters of importance. The parameters are computed for a given experimental condition.

where $GrowthEventRate = \frac{Total\ Length\ grown\ in\ \Delta t}{\Delta t}$

The mean and standard deviation (μ, σ) of the dynamicity of all the microtubules belonging to one experimental condition are defined as:

$$\mu_{Dynamicity} = \frac{1}{N_{MT}} \sum_{j=1}^{N_{MT}} MTDynamicity(j) \tag{2.5}$$

$$\sigma_{Dynamicity} = \frac{1}{N_{MT}} \sum_{j=1}^{N_{MT}} (MTDynamicity(j) - \mu_{Dynamicity})^2 \tag{2.6}$$

$MTDynamicity = \frac{Total\ MT\ length\ increased\ in\ T + Total\ MT\ length\ decreased\ in\ T}{T}$ for a MT.

In this chapter, a detailed overview on how microtubule studies are currently carried on in biology laboratories was presented. First, a detailed discussion was given on the structure and function within cells. The main steps prior to having a microtubule video are: a) sample preparation and b) microscopy imaging of dynamics. Example research work on microtubule dynamics were discussed to illustrate the current questions of importance biologists try to answer. Finally, the main parameters describing microtubule dynamics used in microtubule studies were explained.

Chapter 3

Automated Detection and Tracking of Microtubules: Challenges

An essential step in most microtubule dynamic studies is the localization and tracking of microtubules, which is generally performed manually. The tracking data is then used to derive several dynamic parameters as discussed in chapter 2. The most important pre-processing step is that of extracting suitable features for microtubule tracking, in this case we use microtubule tips. The main contributions in this chapter are: i) automated extraction of the cell region corresponding to microtubules of interest for tracking, and b) enhancement and microtubule tip detection.

The rest of the chapter is as follows. Section 3.1 presents a detailed overview on the literature of automated microtubule detection and tracking, as well as relevant work in active contour and curvilinear structure tracking. Section 3.2 gives details on our mi-

crotubule feature extraction. This includes microtubule layer extraction, enhancement and tip detection. Finally, a summary of the chapter is given in Section 3.3.

3.1 Related Work

The manual tracking of microtubules is a time consuming, and laborious task; it can involve unintentional bias. The number of microtubules being tracked per cell is small, typically about 5 microtubules per video. Hence the complete dynamic behavior of microtubules in a cell is not fully captured. Generally, computation of microtubule length is based on the Euclidean distance between a fixed reference point and the tracked tip. The Euclidean distance can lead to inaccurate length estimates when the microtubule structure is not linear, as shown in Fig. 3.1. Additionally, since the body of the microtubule is not captured, no inferences can be made about shape changes under different experimental conditions.

Beside automating the manually performed tasks, improved detection and tracking would be expected to facilitate new analyses that are not currently feasible with the existing manually tracked results. For example, accurate localization of the entire microtubule enables shape computations and studying neighboring effects of the microtubules in their growth and shortening.

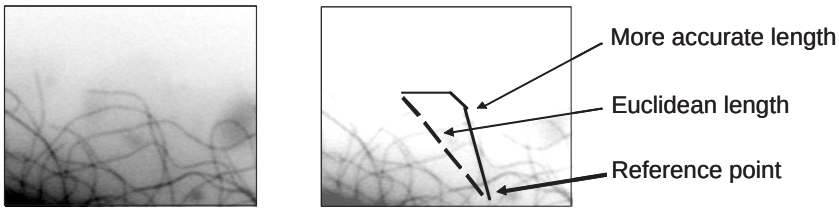


Figure 3.1: Right: an image showing the two type of length computations: i) current length computation using a Euclidean distance between the microtubule tip and a reference point, and ii) the more accurate distance along the microtubule body (the image brightness is increased for clarity). The original image is shown on the left.

3.1.1 General Tracking Approaches

Automated tracking of objects has a rich history in computer vision research [29], [34], [36], [37], [68], [93], [105]. Tracking algorithms can be broadly classified as model-based and non-model based techniques. Model-based tracking assume a model of the tracked object (either 2D or 3D) and formulate the problem of tracking as one in which the model parameters are estimated for each frame. An area where extensive use of object models is human tracking [93], [29]. On the other hand, non-model based techniques are based on the idea of matching regions from frame to frame using visual features such as color, texture and motion. Examples include [37], where the authors merge regions from two adjacent frames using motion and spatial similarity, starting with over segmented frames. In [105], authors use the concept of dynamic motion layers, which produces several independent motion layers in the sequence. Layer parameters and layer membership are estimated simultaneously using a generalized EM

algorithm. In [68], multiple cues such as motion and color, are used in segmenting the video frames. Class labels are computed for video pixels using a maximum likelihood (ML) approach, and a set of weights for the ML function are adjusted using a confidence measure on the extracted optical flow. In [34], tracking uses ideas from model-based and non-model-based approaches. Edge features are used in tracking along with a pre-defined object model.

3.1.2 Active Contour Tracking

Active contours are deformable curves that deform in the image plane according to image features and internal smoothness constraints. In general, active contours are of two types: parametric active contours and geometric active contours. Parametric active contours or snakes were developed by Kass et al. [67]. Parametric active contours define a curve on the image plane using an explicit parametric contour representation that deforms under external and internal forces. Denoting the curve as $C(s)$ where $s \,\epsilon\, [0, 1]$ is the curve parameter as shown in Fig. 3.2. Note that in the case of a closed curve $C(s)$, we have $C(0) = C(1)$.

In the active contour formulation, $C(s)$ is allowed to deform under a balance of forces to minimize the following energy functional:

$$E(C(s)) = \int_0^1 \frac{1}{2}\left(\alpha \left|C'(s)\right|^2 + \beta \left|C''(s)\right|^2\right) + E_{ext}\left(C(s)\right) ds \tag{3.1}$$

where $\frac{1}{2}\left(\alpha \left|C'(s)\right|^2 + \beta \left|C''(s)\right|^2\right)$ is the internal snake energy composed of a balanced

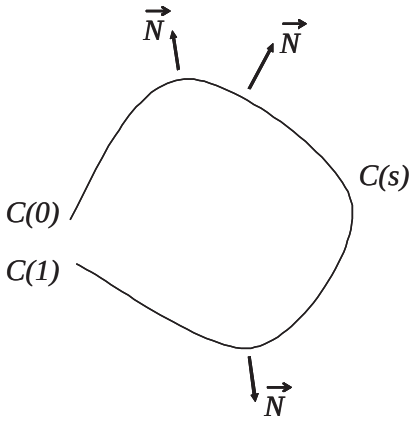


Figure 3.2: A curve $C(s)$ expressed in parametric form with $\overrightarrow{N}$ being the outward curve normal at any given point. This curve is used as an active contour deforming under external and internal forces.

weight of tension and rigidity of the contour. The tension and rigidity of the contour are expressed by $C'(s)$ and $C''(s)$, the first and second order derivative of the contour, respectively. Both derivatives are computed with respect to the curve parameter s. The external energy $E_{ext}(C(s))$ represent image features that we want the contour to snap on such as image edges. The deformation of an initial snake is carried out using principles from calculus of variations and gradient descent minimization as in [67], or using dynamic programming as in [9]. Snakes can be open or closed curves and the parametric representation of the curve can be directly corresponding to the snake points [112], or in a form of B-spline coefficients of the contour segments [104]. In the area of object detection and image segmentation there has been numerous proposed parameter active contour models each of which defines an energy functional that deforms a curve on the image so as to converge finally on the features of interest of the object (such

as image edges). Example parametric active contour approaches include the original snakes work [67]. In [25], a balloon force is added to the traditional image-based external forces ($-\nabla E_{ext}(C(s))$, so that the resulting total external force F acting on the contour is of the form:

$$F = \lambda \overrightarrow{N} - \nabla E_{ext} \tag{3.2}$$

where ∇E_{ext} is the gradient of the external force and $\overrightarrow{N}$ is the outward curve normal at any given point on the curve (See Fig. 3.2). Depending on the sign of λ, the balloon force can expand (for positive values) or shrink the curve $C(s)$ in the normal direction with a magnitude $|\lambda|$. The main motivation behind the balloon force is to allow evolution in smooth image areas and to avoid the requirement that the initial curve be very close to the desired image boundaries as in the original snake formulation.

A more robust snake model was developed based on a specially designed vector force (gradient vector flow) in [112]. It was shown that the vector field has a larger capture field to object boundaries compared to simple gradient and that it is capable of reaching concave regions. The disadvantage of parametric active contours is that change of topology of the contours are not handled naturally, special care need to be carried out to check for splitting and merging of curves such as in [28].

Geometric active contours represent the contour in an implicit form as a cross section of a high dimensional embedding surface as in [88]. We follow the notation from [70] in describing the basic idea behind the level set method. Let us define the

horizontal cross section of a function $U(x, y)$ by the set of points for which $U(x, y) = a$, where a is any constant. For a 3-D function $U(x, y)$, the horizontal cross section is the intersection of a horizontal plane at height a and $U(x, y)$, as shown in Fig. 3.3.

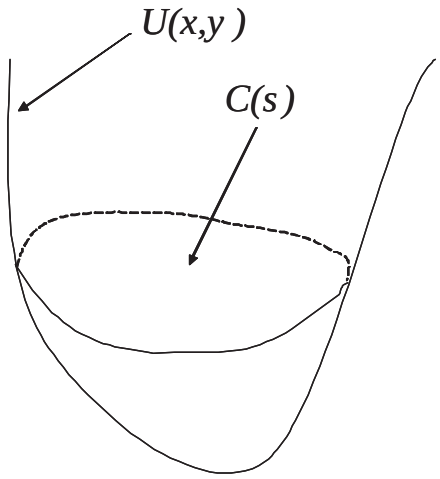


Figure 3.3: Illustration of the concept of representing a curve $C(s)$ implicitly as the level set of function $U(x, y)$.

The set of points for which $U(x, y) = a$ is termed the a^{th} level set of U. Without loss of generality, we will use for this discussion the zero level set of U. The main idea behind the level set method is to represent the active contour in an implicit form as being the zero level set of a function U as shown in Fig. 3.3. Mathematically, the curve $C(s)$ is defined as:

$$C(s) = \{(x, y)|U(x, y) = 0\} \tag{3.3}$$

The curve deformation under external and internal forces is now implemented on

the embedding surface U, rather than the curve $C(s)$. This Eulerian formulation - as opposed to the explicit Lagrangian formulation- has the advantage that topological changes of the deforming curve are naturally handled, with the capability of handling singularities such as corners. The zero level set of a deforming surface U can evolve from being a simple closed curve to multiple simple curves as shown in Fig. 3.4.

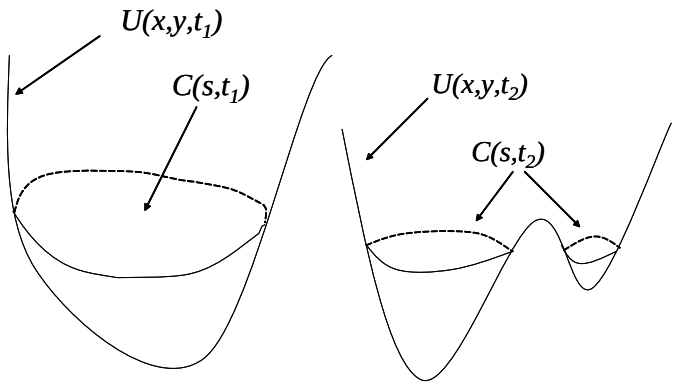


Figure 3.4: Two time instants, t_1 and t_2, in the evolution of a curve. The curve is represented implicitly as the level set of a function U. Note that the curve started as one simply connected curve at time t_1 and later split into two simple curves. This illustrates the power of the level set method in handling topological changes in the active contour.

However existing level set methods for curve evolution are hard to apply to open curves and require intersection of level set functions which can be more computationally expensive than using parametric active contours [89]. Example work in the literature of object segmentation for geometric active include the geodesic active contour

model [18]. A geodesic is simply the minimum path on a domain between two points. In the case of a uniformly weighted flat domain, the geodesic between two points is a line. In [18], the active contour is deformed until it achieves the minimum of the functional:

$$\int_0^1 g\left(C(s)\right)\, ds \tag{3.4}$$

Thus, the weighted contour length ($g(C(s))$) is minimized. The weight used in [18] is an inversely proportional function of the image gradient driving the contour to settle on region boundaries:

$$g = \frac{1}{1 + \left|\nabla \hat{I}\right|^p} \tag{3.5}$$

for p = 1, 2 and $\hat{I}$ is a smoothed version of the image I.

Another example of a geometric active contour is the regularized Laplacian zero crossing detector proposed in [69], where the authors interpret classic edge detectors such as Marr-Hildretth and Canny edge detectors in an energy minimization framework. In the literature, fast implementation methods were introduced for the level set approach of curve evolution such as the narrow band [5], fast marching methods [99], and the Hermes algorithm [90].

For the case of microtubule images, we are interested in tracking open ended structures that do not change their topology. Therefore, we choose to use a parametric active contour for capturing the microtubule body.

Active Contour Tracking Approaches

We review the major active contour approaches used for video object tracking. We will classify these approaches based on two features: a) handling large inter-frame motion by motion estimation and b) capability of tracking multiple objects.

It is worth making a note here that level sets approaches can inherently detect multiple objects as curve splitting and merging is natural. However, we do not consider this case to be truly dealing with tracking multiple objects because individual object identities are not preserved when contours evolve. For example consider the case of tracking two objects when one occludes the other partially in time. In such a case a standard level set approach will create one single curve covering both objects losing boundaries between them.

Based on the above two features of active contour tracking approaches, we have four different classes of techniques: i) techniques that do not use motion estimation and track a single object, ii) techniques that do use motion estimation and track a single object, iii) techniques that do not use motion estimation but track multiple objects, and finally vi) techniques that do use motion estimation and track multiple objects.

The simplest case is when a single object/region is tracked with no motion estimation employed as in [55], [90], [104]. In [55], an active contour model based on minimizing a geodesic path using graph searching is proposed to extract boundaries. The initial contour in a new frame is taken from its location in the previous frame. Topological changes - such as multi-object merging need manual intervention. In [104], a

B-spline model is combined with a muti-scale version of the gradient vector flow approach of [112] to delineate boundaries of single objects. Simple frame differencing is used to find contours in new video frames without any explicit motion prediction. In [90], a variational approach is used to detect and track moving objects in image sequences. A geodesic active contour model is employed, consisting of two main components: a) a motion detection, and b) tracking modules. The motion detection term is devised to attract the contour towards moving regions, while the tracking term is designed to drive the contour towards the image edges. As stated in [90], the technique cannot handle large motion as it does not include a motion prediction step between frames. Furthermore, the detection of multiple objects is rather a natural consequence of using level sets without explicit handling of object to object interactions.

The second class of video tracking techniques based on active contours uses motion estimation techniques for tracking a single region composed of one or more objects. The use of motion estimation from frame to frame can be carried separate of the energy minimization process to relocate the contour near the region boundaries in the present frame such as in [19], [42], [76]. The other approaches use motion prediction *within* the energy functional as in [16] which can track multiple occluding objects. However, when occlusion occurs between two objects, the technique in [16] tracks both objects using a single contour. Therefore, the individual object identities are lost.

The third category of active contour video tracking targets multiple objects but does not use motion estimation [85], [120]. In these techniques, the assumption is that the

video frame rate is high enough yielding small displacements betweens frames. In [120], a repulsive interaction between neighboring object contours is introduced. The conventional edge map based on image gradients is replaced by a new edge map that has a repulsive effect between different objects' contours. The contour location in a new frame is simply taken from the previous frame. In [85], an interaction term is introduced in the energy functional that discourages contours of different regions to overlap. Although there is a tracking energy functional different from the one used for the first frame, yet the individual terms assume that the contours in consecutive frames will be close enough because of small displacements and high frame rate.

The last class of video tracking active contour uses motion estimation and addresses the issue of tracking interacting objects. In [63], a system is developed to track laboratory animals using optical flow and refined with a special type of active contours called active rays. Whenever the animals come into contact, a set of heuristic rules are applied to resolve tracking conflicts and assign contours to each animal correctly. The system uses optical flow for motion estimation between frames to cope with large motion.

Our proposed tracking approach falls into the last category since we are interested in tracking multiple interacting microtubules exhibiting large motion between frames.

3.1.3 Detection and Tracking of Curvilinear Structures

Curvilinear structures detection and tracking represent important computer vision problems for applications such as road detection, mammographic image analysis, fiber iden-

tification, fingerprint image analysis and microtubule detection and tracking. Many techniques have been proposed in the literature to detect curvilinear structures such as scale space approaches with Gaussian derivatives [75], the anisotropic Gauss filtering [48], the fusion of two local line detectors followed by a global Markov random field (MRF) [108], using differential geometric properties of images [101] and using active contour approaches [96]. The problem of correspondence of curvilinear structures between sets of images, has also been studied. In the majority of approaches, lines are first detected and then line properties such as orientation, position, width and center lines are used for the matching process [21], [66], [81].

In [21], orientation of lines and position of endpoints are used for matching lines in an image sequence and complemented with a relaxation labelling and temporal grouping processes. In [73], line intensities, diameters and center lines are used in the task of 3D reconstruction. In [66], the authors solve for transformation that map set of lines in an image to another using center points, line lengths and unit vectors along the lines. In [81], the authors use line orientation, width and curvature to match sets of lines in mammograms images taken over a period of time. To track changes in live neurons (morphometry), different models for neuron growth and shrinking are used to label frame-level changes within a Bayesian framework in [7].

In most of these techniques, a threshold is used to binarize the line detector response prior to matching between the images. In the case of noisy and non-uniformly illuminated video sequences, it is very likely that the required threshold will vary from

frame to frame, thus causing a possible loss of the tracked curvilinear structure. Instead of tracking using the binarized response, we propose to use an active contour framework that uses the continuous line detector response to detect and track the curvilinear structure of microtubules. This approach is capable of handling low-signal to noise conditions and non uniform illumination issues in microtubule videos.

3.1.4 Review of Automated Microtubule Detection and Tracking

Automated Microtubule Detection Techniques

Recently, automated microtubule detection has been addressed in [39], [54], [59]–[61]. In the majority of these techniques, a user input is required to mark the ends of the microtubule of interest. An enhancement procedure extracts the microtubule backbone between the two ends. Hence, these techniques were mostly semi-automated in nature. In [39], The proposed technique consists of two main components. First, a tube enhancement filter is used, which defines tube confidence measurement for each voxel. Then, the user manually specifies two voxels, which mark the starting and ending point of the microtubule axis. In this subvolume, the method randomly selects some points, which become points of graph. After that, a graph searching algorithm is applied to these points, and the shortest path based on a specific distance metric corresponds to the central axis of the microtubule. A similar approach is used in [59], the authors propose a model-based 3D image enhancement approach by combining transform domain technique and spatial domain techniques in three consecutive steps. The enhancement

starts with an anisotropic invariant wavelet transform to effectively enhance the elongated features, followed by a 3D shape filter via eigen analysis to capture the local geometric properties of the tubular structure. The enhancement ends with a coherence enhancing diffusion to complete the interruptions along the microtubules.

In [60], the authors propose to automate the microtubule segmentation by extending the active shape model (ASM) in two aspects. First, they develop a higher order boundary model obtained by 3-D local surface estimation that characterizes the microtubule boundary better than the gray level appearance model in the 2-D microtubule cross section. Then they incorporate this model into a weight matrix of the fitting error measurement to increase the influence of salient features. Second, they integrate the ASM with Kalman filtering to utilize the shape information along the longitudinal direction of the microtubules and reduce the effects of noise and clutter. In [61], the authors propose an automated approach to extract the microtubule plus-end with a coarse to fine scale scheme consisting of volume enhancement and plus-end segmentation. To make the segmentation robust against confusing image features, they have fully incorporated the prior knowledge of microtubules and plus-ends into a model-based framework. The automated approach also segments more fine structures that could be overlooked by human operators. In [54], the microtubles are extracted in terms of consecutive segments by solving Hamilton-Jacobi equations. The algorithm extracts the microtubule starting from its tip. Furthermore, a limited 3D reconstruction is performed.

Automated Microtubule Tracking Techniques

In the literature few papers have addressed the automated tracking of microtubules [26], [41], [53]. In [53], the outer tips of microtubules are tracked using an extension of the work in [54]. Streamlines minimizing a cost function are used to find microtubules based on manually entered tips on an initial frame. To the best of our knowledge, the only fully automated technique was presented in [41]. In [41], an initial point is selected on a microtubule and iteratively gives a stack of points representing the microtubule using tangent constraints. Once the microtubule is detected, it is tracked in time while constraining the search space in a normal direction around microtubule points. Visual tracking results are shown for two real sequences but without any quantitative assessment of the tracking performance. Though the tracking algorithm is automated, the authors report difficulties in handling microtubule intersections and do not present quantitative tip tracking performance results for real sequences. The main point with their approach is that microtubule tracking is performed using a local measure of consistency that can lead to problems at intersection.

The work in [26] presents a framework for detecting and tracking diffraction images of linear structures in differential interference contrast (DIC) microscopy. The method requires the user to select few points on the microtubule dividing it into segments. Tracking of the segments is performed using a sum of squared (brightness) differences algorithm.

3.2 Feature Extraction for Microtubule Tracking

Microtubule analysis revolves around computing the microtubule length in every frame. This has been done, manually, by tracking the tip and estimating the length as the Euclidean distance to a fixed point. In order to automate the microtubule tracking for length computation, we first extract a set of suitable features that are tracked over frames. The features used for this purpose are the microtubule tips. The feature detection task is composed of the following modules:

- *Identifying microtubules of interest*: We are usually interested in tracking microtubules near the cell periphery, as most of the microtubule activity occurs at the plus ends away from the cell center. Additionally, the reduced microtubule density at the periphery allows individual microtubules to be resolved. We propose a clustering approach to extract a layer corresponding to microtubules of interest.

- *Microtubule polymer mass detection*: This is the first step towards automated tracking. Given a video frame, we propose a filtering approach to extract a binary mask showing microtubules.

- *Automated microtubule tip detection*: Given the binary mask of microtubules, we propose a morphology-based technique to detect microtubule tips to be tracked subsequently.

3.2.1 Extracting Microtubule Layer by Temporal Clustering

A microtubule video of T frames can be considered as a spatiotemporal volume with the first two dimensions as the spatial frame coordinates (x, y) and the third dimension is the time t, as shown in Fig. 3.5. Let us denote the intensity value at each point in this volume by $f(x, y, t)$. At each pixel location (x, y) on the first frame, we construct a vector $\mathbf{a}(x, y)$ of dimensionality T:

$$\mathbf{a}_{xy} = [f(x, y, 1), f(x, y, 2), ..., f(x, y, T)] \tag{3.6}$$

From $\mathbf{a}_{xy}$, we construct a feature vector $\mathbf{f}_{xy}$ by taking the Discrete Fourier Transform (DFT):

$$\mathbf{f}_{xy}(u) = \left| \frac{1}{T} \sum_{i=1}^{T} a_{xy}(i)\ e^{-\frac{j2\pi iu}{T}} \right| \quad , 1 \leq u \leq T \tag{3.7}$$

where i is the i^{th} and position in the vector $\mathbf{a}_{xy}$. The resulting vector $\mathbf{f}_{xy}$ is of dimensionality T and has as elements the magnitude of the DFT coefficients. The first coefficient $\mathbf{f}_{xy}(1)$ is the average intensity at location (x, y) over the number of frames T. To make the representation intensity invariant, we discard $\mathbf{f}_{xy}(1)$. Furthermore, since there is a symmetry in the magnitude of FFT coefficients, we only take the first half of coefficients. At the end, our feature vector is $\mathbf{p}_{xy}$ (assuming T is odd):

$$\mathbf{p}_{xy} = \left[\mathbf{f}_{xy}(2), \mathbf{f}_{xy}(3), ..., \mathbf{f}_{xy}\left(\frac{T-1}{2}\right) \right], \quad 1 \leq x \leq H,\ 1 \leq y \leq W \tag{3.8}$$

where W and H are the width and height of the video frame respectively. For every (x, y) pixel, we now have a corresponding $\mathbf{p}_{xy}$ value. The motivation behind using a frequency-based representation of the volume is to use it to partition the microtubule video into regions of different spatial activity patterns. We then cluster the different FFT vectors $\mathbf{p}_{xy}$ from all pixel positions using a K-means algorithm into V clusters corresponding to regions of varying activity in the cell. Finally, we extract the clusters corresponding to the region of highest activity, which we call the microtubule layer. An example of the extraction of the microtubule layer is shown in Fig. 3.6.

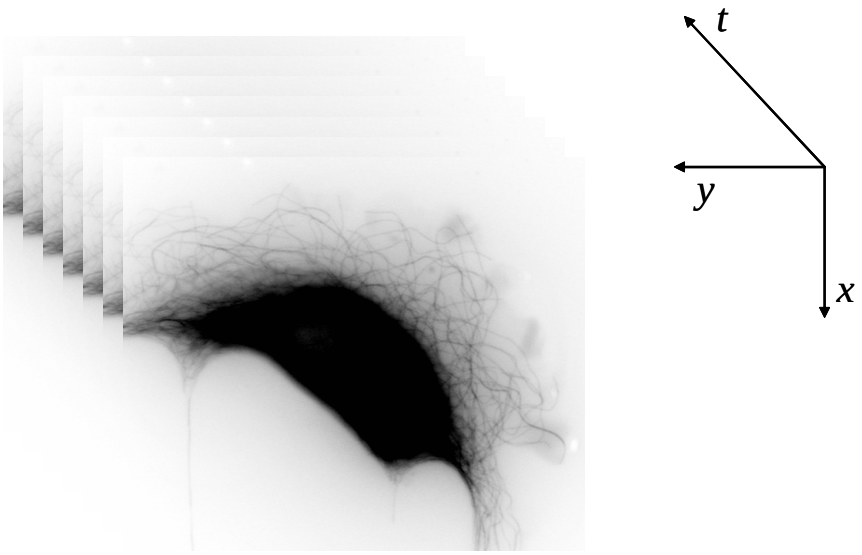
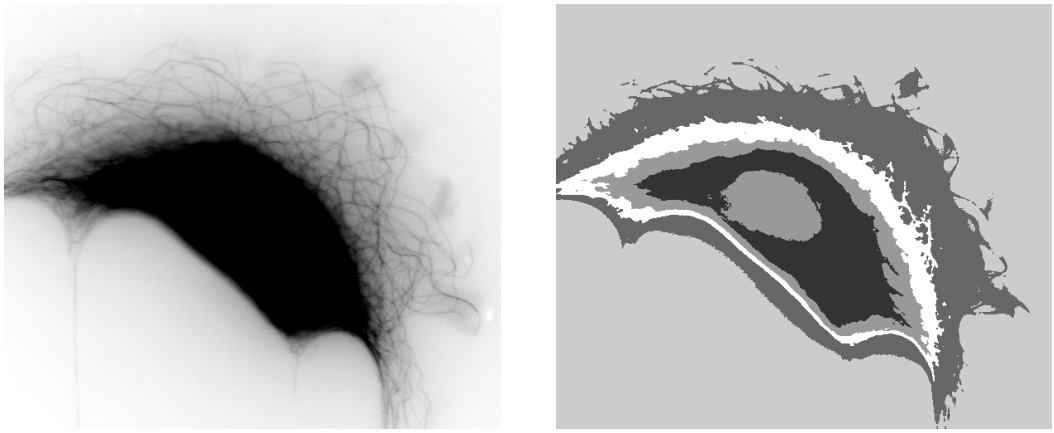
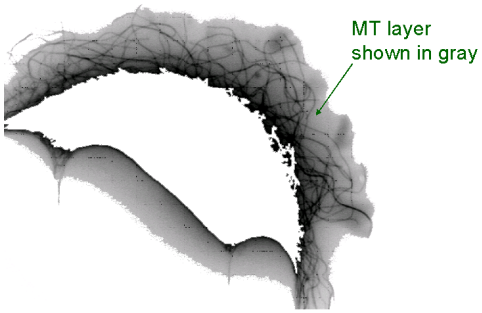


Figure 3.5: The microtubule video represented as a volume.



(a) (b)



(c)

Figure 3.6: Extracting the microtubule layer through temporal clustering. (a) One of the image frames from the video, (b) K-means clustering into 5 clusters including one for the background, and (c) the region of the frame corresponding to the microtubule layer.

3.2.2 Ridge-based Detection of Microtubules

After extracting the microtubule layer, a filtering approach is used to detect the microtubule tips in this layer. Our algorithm for tip detection starts by extracting a binary mask showing the locations of microtubule polymer. The basic assumption about microtubules, that enables extracting a binary mask of microtubule locations, is that microtubules look like black curvilinear structures on a light background in an ideal scenario. A second derivative of Gaussian kernel matched to image locations at different orientations should reveal this tubular structure while eliminating background noise. Let the intensity function in the window of interest W be denoted as I_W, the output after filtering the window is then:

$$I_W^f(x,y) = \max_{\theta}(I_W(x,y) * G''_{\sigma,\theta}(x,y)) \tag{3.9}$$

where $G''_{\sigma,\theta}(x,y)$ is the second derivative of the Gaussian kernel with scale σ. The second derivative is taken along an orientation θ. σ is chosen experimentally based on the microtubule width. Gaussian second derivative kernels for four different orientations are shown in Fig. 3.7. In order to generate a binary image of the microtubule mass while suppressing background noise, we use unsharp masking [50]. We use two thresholds T_1 and T_2:

$$T_1 = \mu_{I_W^f}, \quad T_2 = \mu_{I_W^f} + \sigma_{I_W^f}$$

and

where $\mu_{I_W^f}$ and $\sigma_{I_W^f}$ are respectively the mean and the standard deviation of the filtered window I_W^f. We compute the binary images B_1 and B_2:

$$B_1 = sgn(max(I_W^f - T_1, 0))$$

and

$$B_2 = sgn(max(I_W^f - T_2, 0))$$

where sgn is the sign function. Finally, the binary microtubule mask B is obtained:

$$B = sgn(B_1 B_2)$$

This way we essentially preserve the details of thresholding at the true microtubule regions, while suppressing noise on the background. An example of finding the maximum of a second derivative of Gaussian convolved with the image at all pixel locations, with σ = 2 and eight different orientations, is shown in Fig. 3.8, Fig. 3.9 and Fig. 3.10.

3.2.3 Detecting Microtubule Tips

The binary microtubule mask computed based on second derivative of Gaussian filtering is thinned to generate one pixel width lines. A test is performed at every white pixel

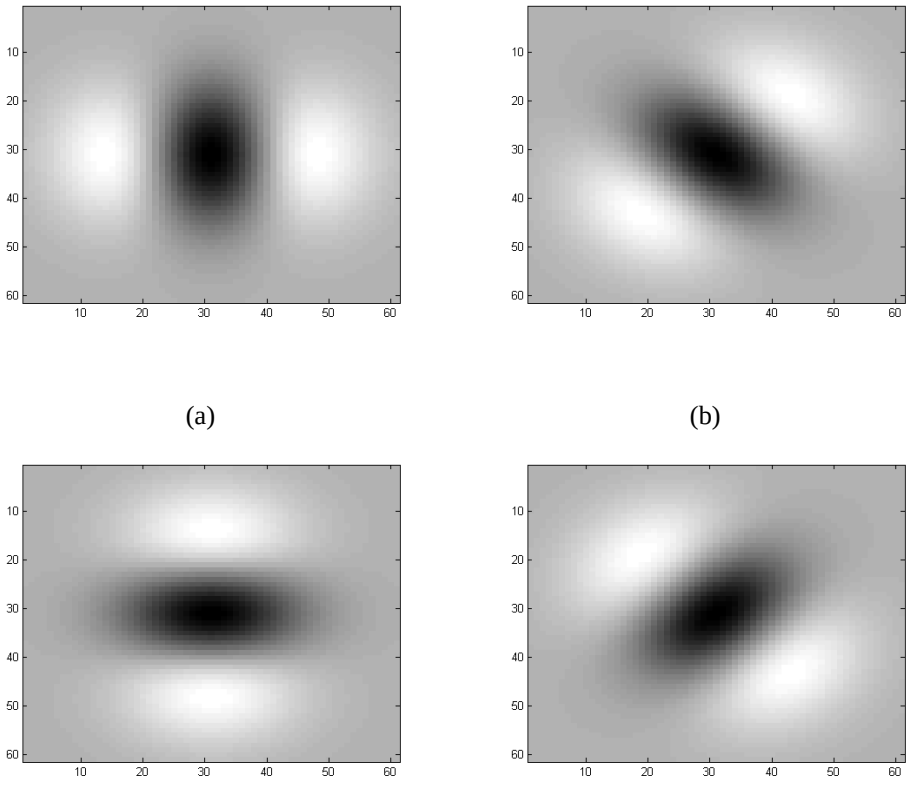
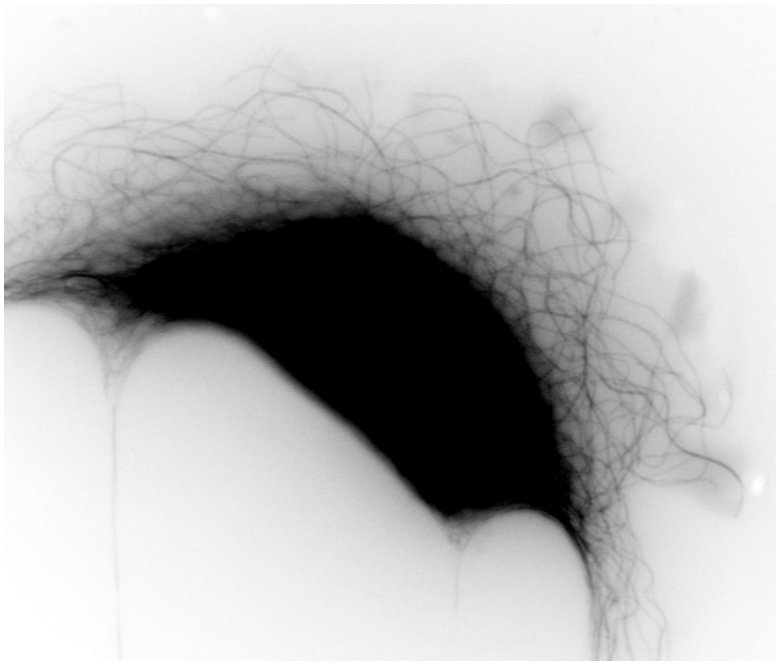
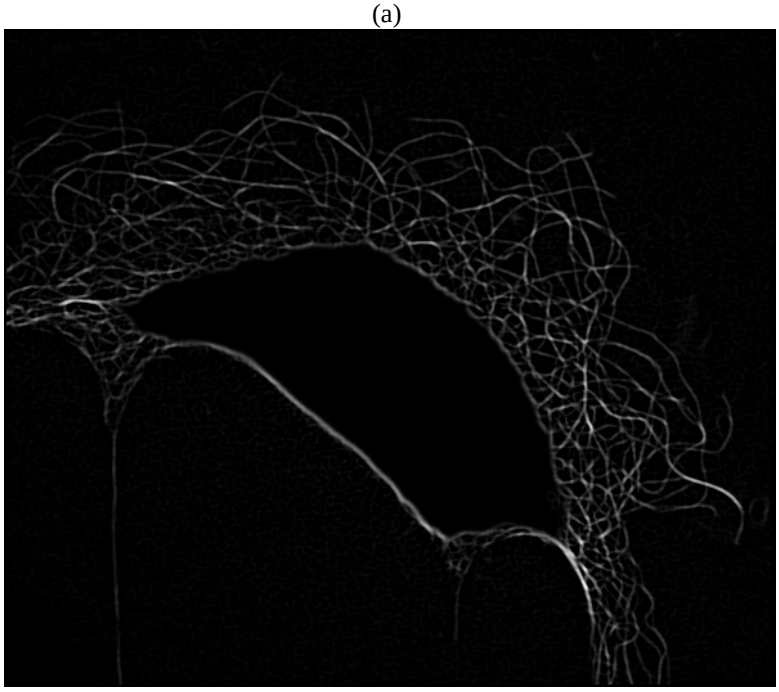


Figure 3.7: Gaussian second derivative kernels for (a) $\theta = 0°$, (b) $\theta = 45°$, (c) $\theta = 90°$ and (d) $\theta = 135°$.

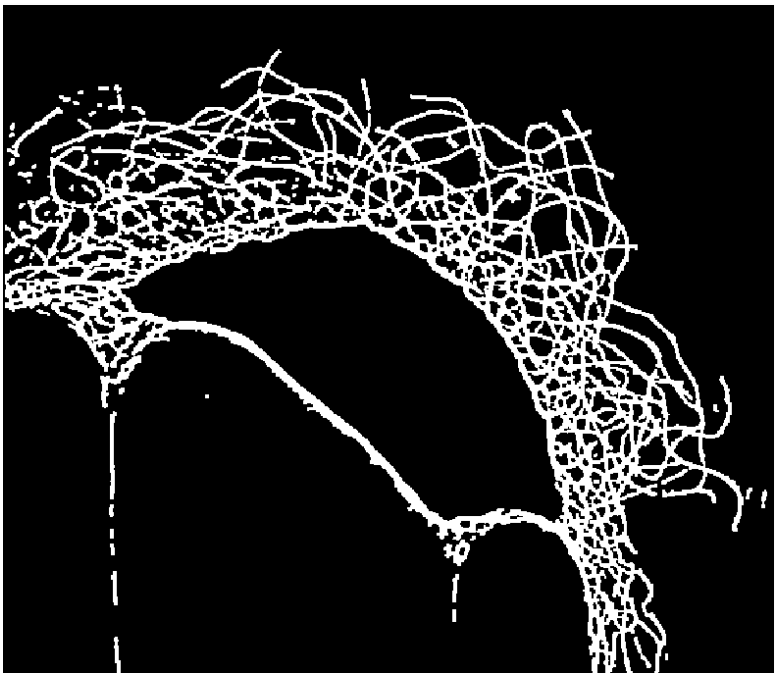


(a)

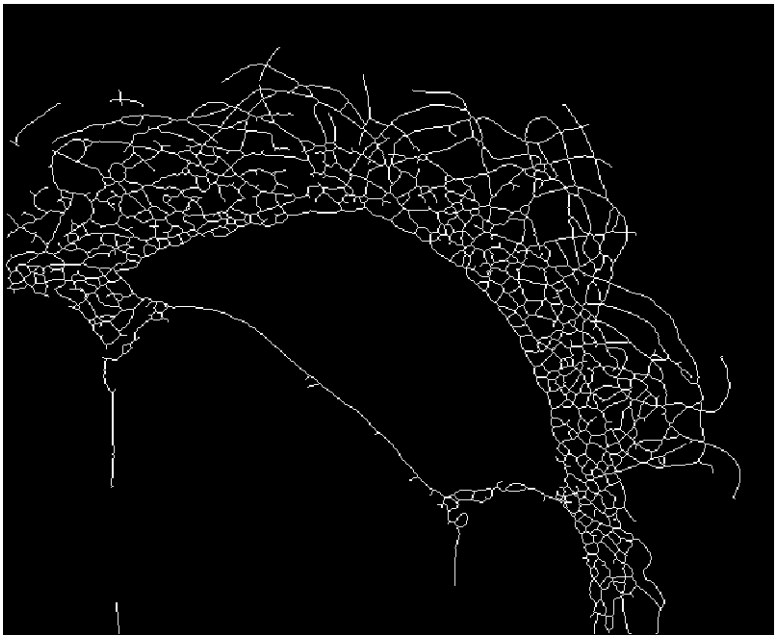


(b)

Figure 3.8: (a) A microtubule video frame and (b) the second derivative of Gaussian filter output.



(a)



(b)

Figure 3.9: (a) Binarization of the filter output shown in Fig. 3.8 and (b) thinning of the binarization used as input to the tip detector.

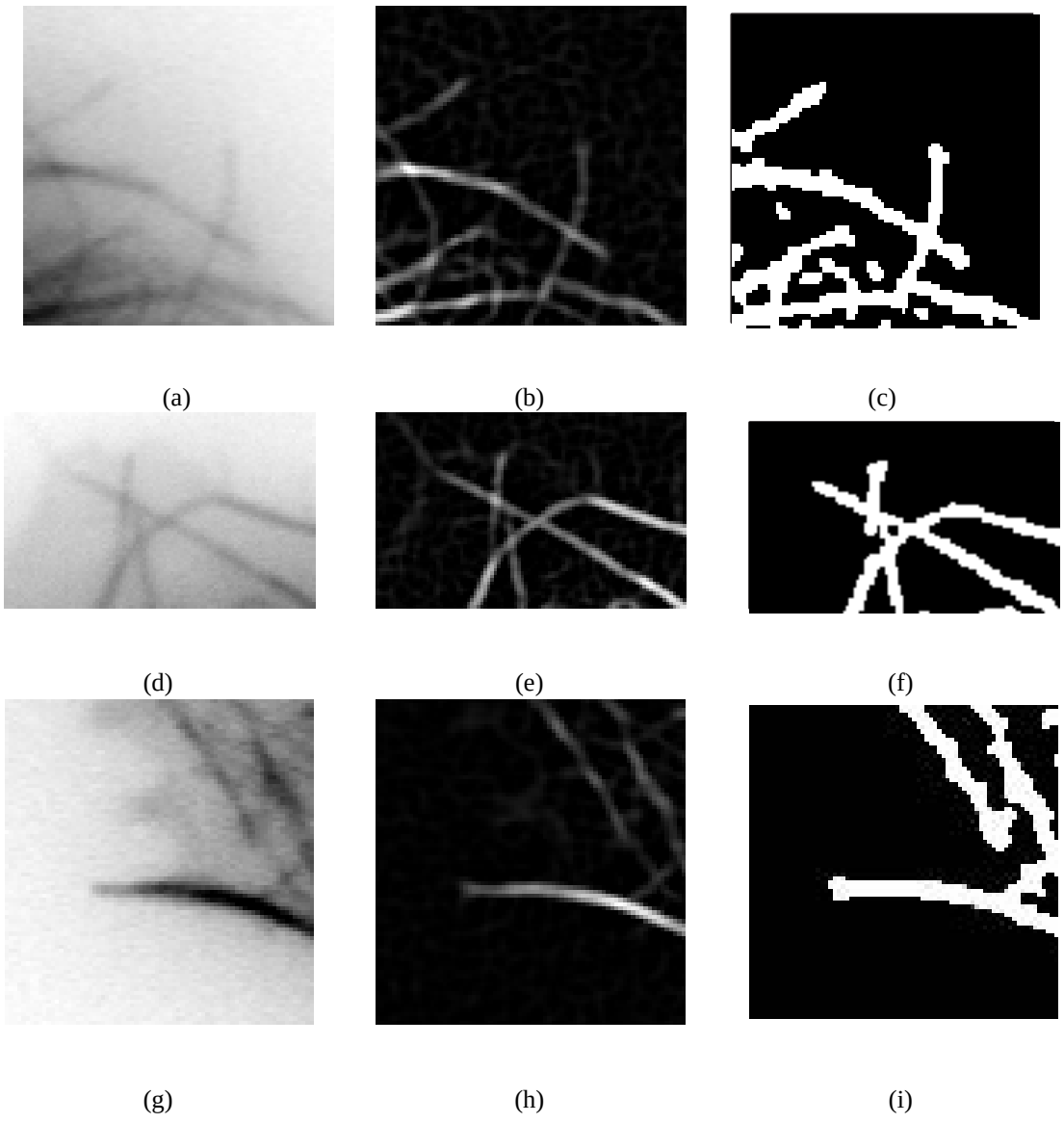


Figure 3.10: Left column: window I_W in a video frame, middle column: filter output I_W^f and right column: binarization.

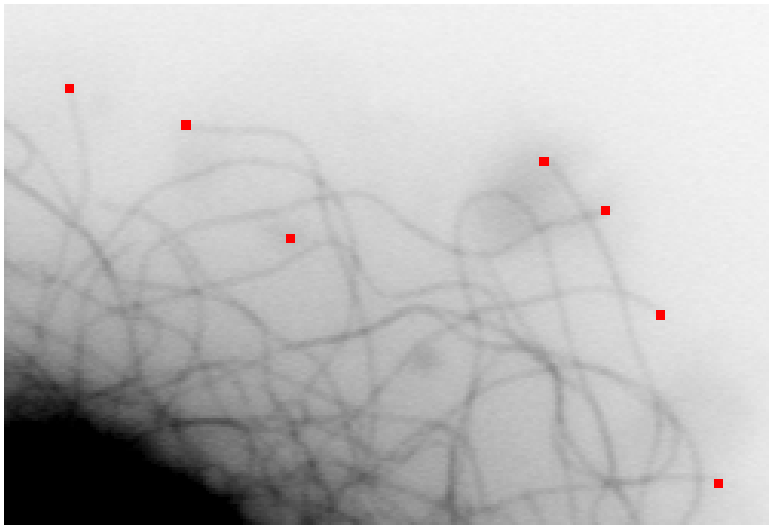
location to check if the pixel is a line ending. Mimicking the manual method to select microtubule tips that are free of too much clutter, we filter out tips that are in locations where the ratio of microtubule polymer to non-polymer masses is less than a threshold - we take it 0.3 experimentally. Examples of microtubule tip detection for two video sequences are shown in Fig. 3.11 and Fig. 3.12.

Quantitative Evaluation of Tip Detection

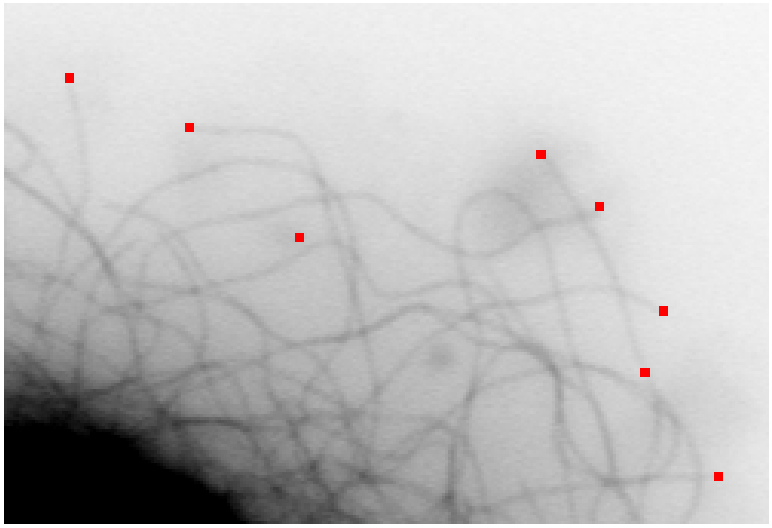
We are using six microtubule videos as the basis of the tip detection performance. In each video, we manually marked the microtubule tip locations in the 6^{th} frame. We ran our tip detector on the 6^{th} frame of each video sequence. Consider the case that we detect N_1 tips, while there are N_2 manually selected tips. To evaluate the performance of the tip detector, we use a bipartite graph matching, with the two sets of tips N_1 and N_2 being the two parts of the graph, to compute the precision recall of the tip detector. Consider the case that out of the N_1 detected tips, there are N_1' true tips. The precision P and recall R values of the tip detector in this case are as follows:

$$\begin{aligned} P &= \frac{N_1'}{N_1} \times 100\% \\ R &= \frac{N_1'}{N_2} \times 100\% \end{aligned} \tag{3.10}$$

We have a parameter in the tip detector that controls the number of detected tips. Consider that we run the tip detector on a frame and obtain N frames. For each tip, out of the N tips, we compute a test value T_{perc} in a window Win of size 61 $\times$ 61 around the

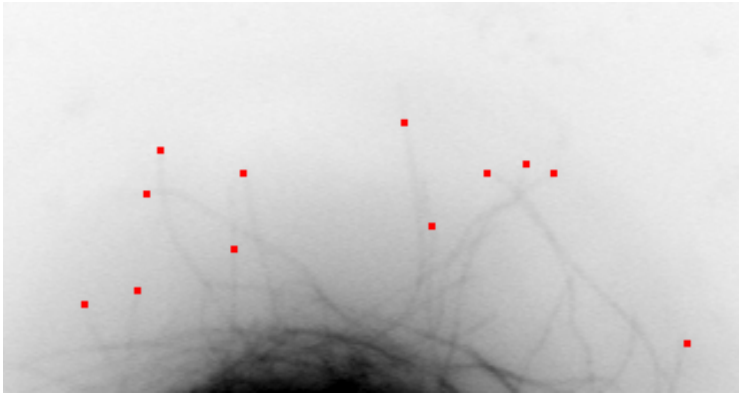


(a)

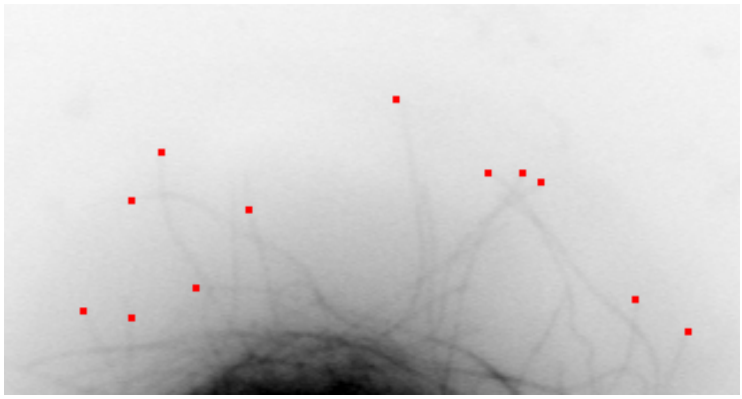


(b)

Figure 3.11: Examples of microtubule tip detection in two consecutive frames of a video sequence.



(a)



(b)

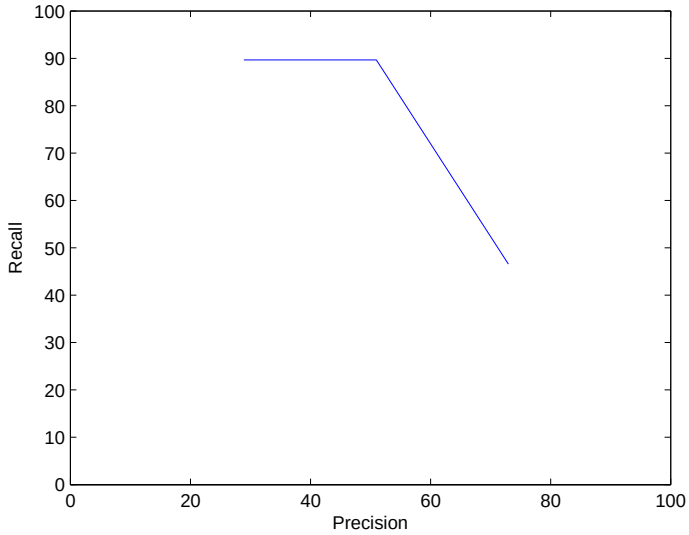
Figure 3.12: Examples of microtubule tip detection in two consecutive frames of a video sequence.

tip. The test value in the window is the ratio of polymer mass to the total size of the window. Computationally, this test value is the percentage of white pixels in the binary microtubule detector image within the window Win. Consider a threshold value th_1, if T_{perc} within the window around a tip is greater than th_1, then we reject the tip, otherwise we consider it as a tip. The main motivation behind this filtering is that humans usually select tips in the peripheral region of the cell, where the test value T_{perc} is likely to be low. By varying the threshold th_1, we obtain a variable number of tips in the considered frame. Thus, with the manually marked tips, we can now compute a precision-recall curve for the tips within the frame by varying the threshold th_1. For each value of th_1, we have a unique precision-recall value for a given frame. We take the following values for th_1: 0.2, 0.4, 0.6, 0.8 and 1. To summarize the precision-recall value at each th_1 value, we use the F-measure [95] with equal weighting of the precision recall values defined as follows:

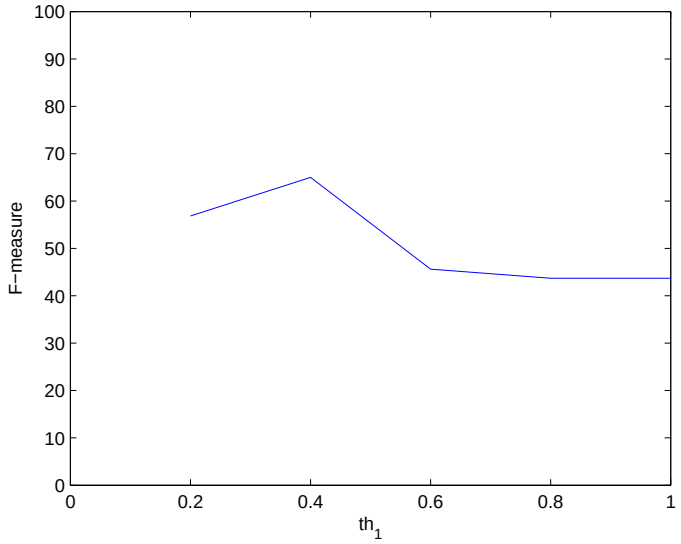
$$F = \frac{2PR}{P+R} \tag{3.11}$$

Example precision recall curves and their corresponding F-measures for two frames in two example microtubule videos are shown in Fig. 3.13 and Fig. 3.14.

By computing the F-measure over the 6 frames corresponding to frame number six in each of the microtubule video sequences, we found that the optimum value of T_{perc} is 0.2. With the value of 0.2, we show in Fig. 3.15 the F-measure of all the videos.

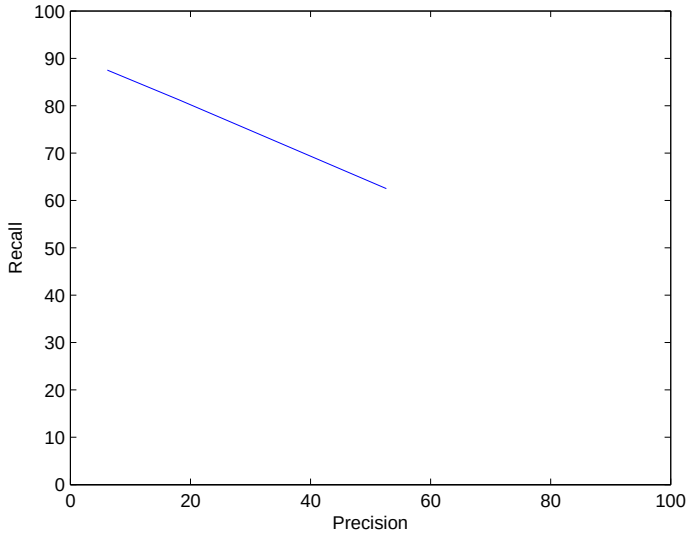


(a)

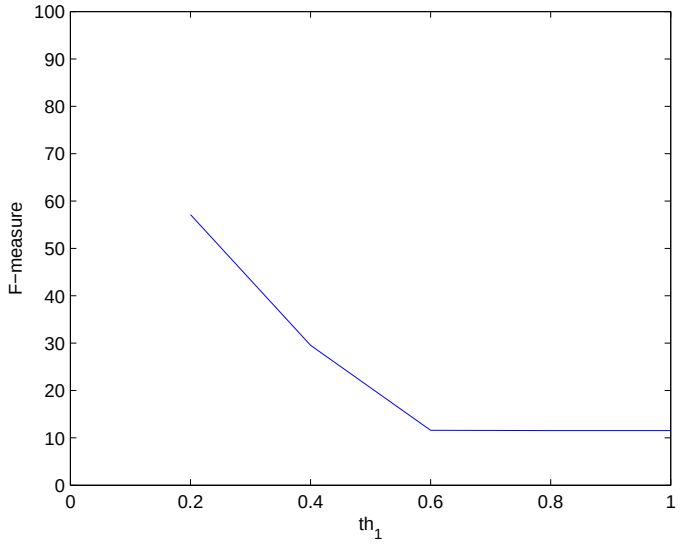


(b)

Figure 3.13: (a) Precision-recall of tip detection in the sixth frame of a video sequence and (b) its corresponding F-measure for equal weighting of precision and recall.



(a)



(b)

Figure 3.14: (a) Precision-recall of tip detection in the sixth frame of a video sequence and (b) its corresponding F-measure for equal weighting of precision and recall.

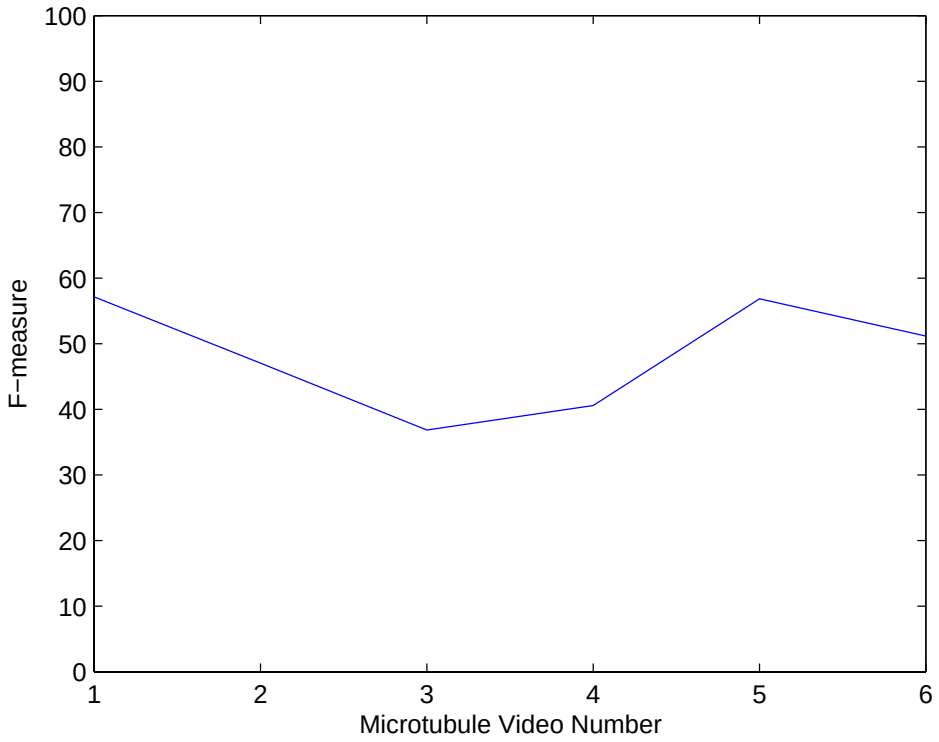


Figure 3.15: The F-measure of the tip detector in the sixth frame of the 20 video sequences.

3.3 Summary

We have discussed in this chapter the problem of automated detection and tracking of microtubules. First, we presented a detailed review of the state-of-the-art microtubule detection and tracking in the literature. We also discussed relevant related work on tracking curvilinear structures, active contour tracking approaches as well as general tracking approaches. Second, we introduced the preprocessing approach that we are developping in this thesis for tracking microtubules. Specifically, we looked into the automated extraction of microtubule tips, used as features for tracking in Chapter 4. Automated extraction of microtubule tips is solved using three modules: a) microtubule layer extraction, to extract the microtubules of interest for tracking, b) filtering and binarization of microtubule polymer mass and c) detecting microtubule tips from the binarization, using a morphological approach. We have shown example of visual and quantitative experimental results for tip detection. With the microtubule tips detected in every frame of the video, we now have the features used for tracking in the next chapter.

Chapter 4

Spatio-temporal Tracking of Microtubules in Live Cell Imaging

Automated tracking of microtubules could provide critical insights into understanding the molecular basis of microtubule dynamics. In Chapter 3, we have proposed a technique to extract microtubule tips to be used as features for tracking. Based on these features, we introduce in this chapter a novel automated technique for microtubule tip tracking. The proposed tracking algorithm is based on a spatio-temporal tip matching over video frames to form tracks. We formulate the problem of tip matching over frames in a graph-based context. Hence, tip matching becomes a maximum matching on the graph, for which there are efficient numerical implementations. Graph vertices are the microtubule tips and edge weights correspond to similarity between tips. After computing the best matching tips, we proceed to form the microtubule body using a

Fast Marching technique [99]. Microtubule body deformations are addressed using an active contour approach. The proposed spatiotemporal tracking algorithm addresses the following challenges in tracking microtubules:

- Microtubules appear as tubular structures in the image frames. The shape of microtubules vary within the same cell widely. An accurate estimation of the length of the microtubule should take into account the curvilinear structure. The currently widely used manual tip tracking method and statistical methods ignore this effect.

- Microtubules undergo large changes in length from frame to frame because of growth or shortening at either microtubule ends. The large change in length, together with frequent occlusions, pose significant problems to any tracking method.

- The images of microtubules have low signal-to-noise ratio and exhibit nonuniform illumination both spatially and temporally.

The rest of the chapter is as follows. Section 4.1 gives an overview of the proposed spatio-temporal microtubule tracking technique. In Section 4.2, we give details on solving the tip matching problem in a video. We discuss the formulation of the tip tracking problem in a graph-based context, the similarity metric suggested as weights on the constructed graph and the algorithm used to compute the maximum matching over the graph. In Section 4.3, the algorithm for microtubule body formation from tips and the proposed active contour for handling lateral motions are presented as well as

visual and quantitative experimental results. Finally, a summary of the chapter is given in Section 4.4.

4.1 Multiframe Tracking Approach

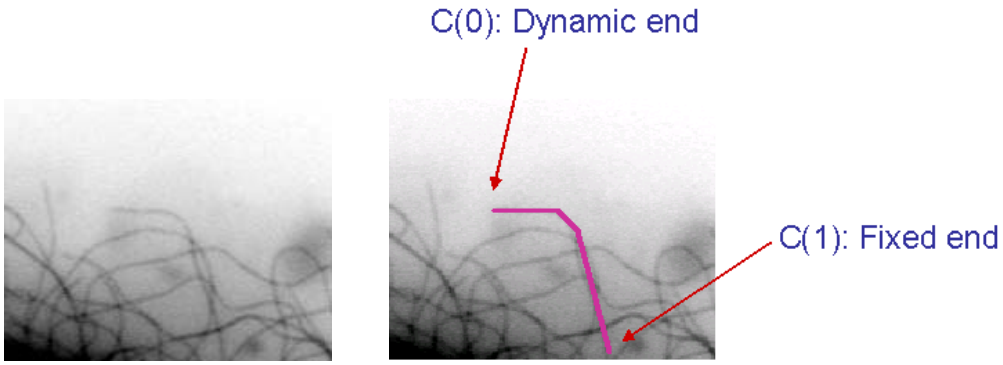
We model a microtubule in a video frame as an open curve $C(s)$ where $s \epsilon [0, 1]$ is the curve parameter (see fig. 4.1). Due to the highly dynamic nature of microtubules and their frequent intersections, a spatiotemporal tracking algorithm based on deformable contours is an effective strategy. We propose to use an active contour based on line features to capture deformations of the microtubule in a given frame. The active contour is an open curve in this case with one end fixed and the other - the tip- is free to move. Tracking the microtubule in a frame is performed by finding the image location which minimizes a functional $E(C(s))$ over the image plane:

$$E(C(s)) = \int_0^1 (E_{\text{int}}(C(s)) + E_{ext}(C(s)))\, ds \tag{4.1}$$

with: $C(0)$ is the microtubule end, assumed fixed over frames, and $C(1)$ is the microtubule tip free to move.

The multiframe approach suggested here is based on the following modules:

- *Tip detection*: Given a microtubule video, we design a tip detector generating the locations of microtubule tips in every frame. Note that the number of the detected tips in every frame need not to be equal.



(a) (b)

Figure 4.1: Microtubule model as an open curve with only one moving end – the tip.

- *Multiframe tip matching to generate tracks*: Given the detected tips in every frame, tips need to be matched between frames to form microtubule tracks. This is formulated as a graph matching problem, by considering all tips from all the frames simultaneously. In this way, this formulation is capable of tracking multiple interacting microtubules. The advantage of multiframe tip matching is that it enforces continuity of tracks, is able to handle missing tips in some frames due to noise, and is able to discard false positives.

- *Extracting microtubule bodies*: An active contour-based approach is proposed to track the full body of the microtubule based on the tip locations in a given microtubule track. Being able to track changes in the full body of the microtubule instead of just the tip location enables a better estimate of the microtubule length. Furthermore, shape changes can now be studied in microtubules which can not

be performed based on tip tracking only.

The block diagram of the tracking technique is shown in Fig. 4.2. Tip position estimation in any given frame has been discussed in Chapter 3 (Section 3.2.3), the spatiotemporal matching of tips is now detailed.

4.2 Spatiotemporal Tip Matching

Having generated a set of tips in every frame of the microtubule video, corresponding tips are matched between frames to form microtubule tracks. The matching is considered from all the frames directly in a spatiotemporal manner. At the end of the tip matching over all the frames, the longest tracks are selected for further processing, since short tracks are likely to be due to noise. It is worth noting that the computed tracks in this manner can start and end at any given frame of the video sequence, can have arbitrary lengths, and can be skipping frames in the middle. Tip matching is formulated as a graph matching problem.

4.2.1 Graph-based Formulation

Consider a microtubule video of length T frames. Let us denote N_i to be the number of tips detected in frame i for $1 \leq i \leq T$. Denote the tip detected in a frame as t_i^h with the subscript corresponding to the frame number and the superscript corresponding to the tip number in frame f_i, thus h has the range $1 \leq h \leq N_i$. We construct a

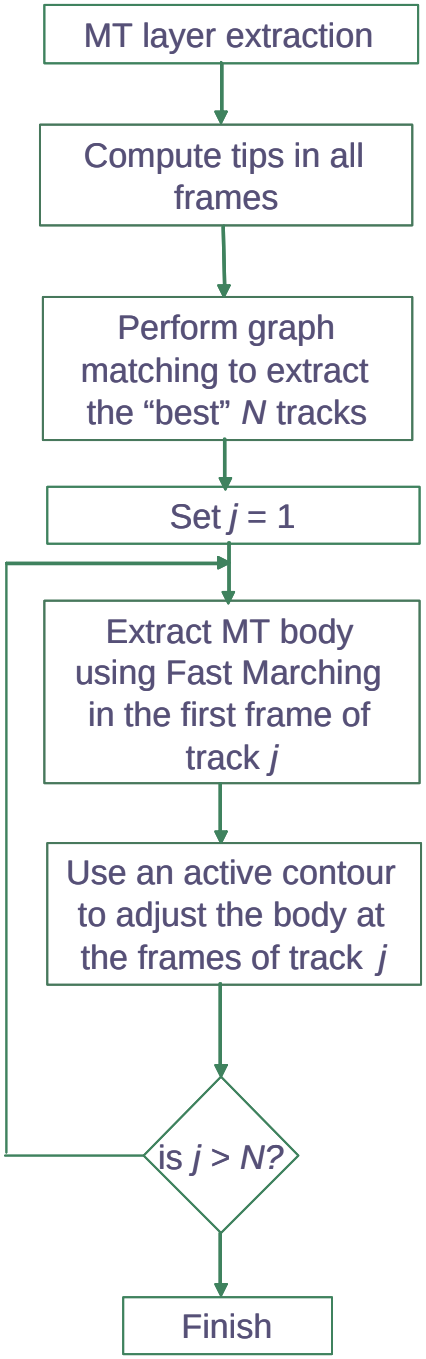
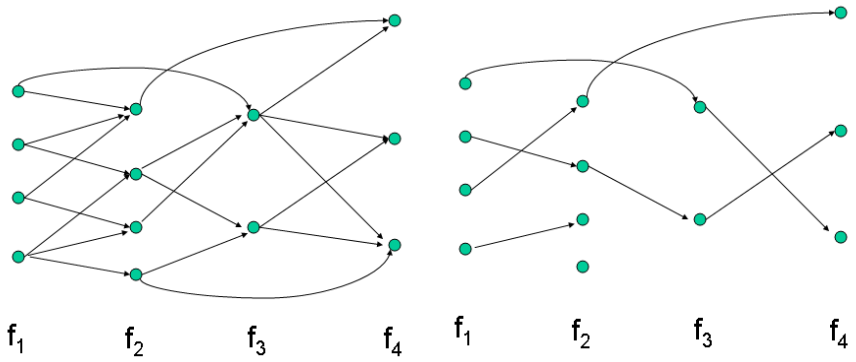


Figure 4.2: The block diagram of the proposed spatiotemporal tracking technique.

graph $G = (V, E)$ whose vertices V correspond to the detected tips in every frame and the edges E represent similarity between vertices. The edge weights of the graph represent the matching gain of corresponding two tips in different frames. In order to allow microtubule tracks to skip some frames, edges are included between tips in non-consecutive video frames. An example of a graph used for tracking microtubules is shown in Fig. 4.3 with a possible solution of tip matching.



(a) (b)

Figure 4.3: (a) An example graph whose vertices are the tips detected in every frame of video (here shown for a length 4 video) and (b) a possible maximal matching solution. The attributes at each vertix are the frame number and the tip number in this frame. Note that the tracks can be of different lengths, start and end at arbitrary frames, and skip frames in between.

The main advantages of formulating the tip matching problem using a single graph encoding all the tips from all the frames are:

- It can handle missing tips due to noisy conditions by allowing the final micro-

tubule tracks to skip frames in between.

- It can potentially remove spurious tips found if the noise/loss in signal does not occur repeatedly at nearby locations.
- It can handle tip occlusion, since the graph matching allows tips in non-consecutive frames to be matched.

4.2.2 Similarity Metric

The main metric which defines the matching of tips for the video frame is the similarity measure linking tips in different frames. Consider two tips t_i^h and t_j^r in two separate frames f_i and f_j. For example, one can use:

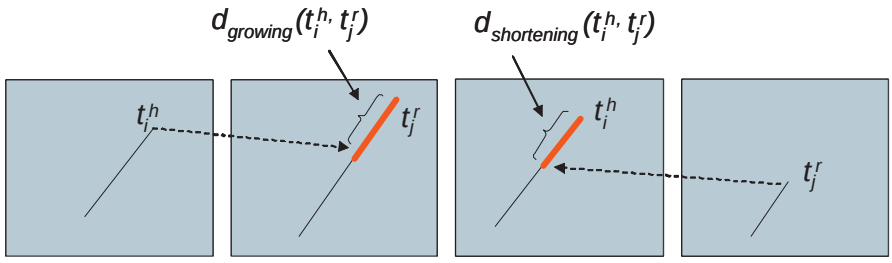
$$Sim(t_i^h, t_j^r) = \frac{1}{1 + d(t_i^h, t_j^r)} \tag{4.2}$$

where $d(.,.)$ is the Euclidean distance between the two tips. However this will have problems in cases of tips of different microtubules coming close to each other. A better alternative is to consider a distance constrained on the microtubule body, such as a geodesic distance. The geodesic distance assumes the knowledge of whether the microtubule is growing or shortening. Since, we do not have a-priori information on growth or shortening of a microtubule, we consider both cases between two different frames as in Fig. 4.4. Let us denote the geodesic distance computed with the assumption that the microtubule is growing as the first geodesic distance $d_{growing}$. Similarly, we denote the geodesic distance in the case of shortening as the second geodesic distance $d_{shortening}$.

For the microtubule growth, we project the location of the tip t_i^h on frame f_i to the same location on frame f_j. We compute $d_{growing}(t_i^h, t_j^r)$ as shown on Fig. 4.4.a. For the microtubule shortening case, we back-project the location of the tip on frame f_j to the same location on frame f_i. We then compute the geodesic distance $d_{shortening}(t_i^h, t_j^r)$ as shown on Fig. 4.4.b. Finally the similarity metric (edge weight on the graph G) used between the two tips t_i^h and t_j^r is computed as follows:

$$Sim(t_i^h, t_j^r) = e^{-min(d_{shortening}, d_{growing})}, \tag{4.3}$$

where the exponent is taken as the minimum of the two geodesic distances.



(a) (b)

Figure 4.4: Illustration on how the similarity weight between vertices of the graph is computed between tips in two different frames f_i and f_j (see text for explanation). (a) Case of growing and (b) shortening.

4.2.3 Maximum Matching on the Graph

Given the graph G of microtubule tips as vertices and the edge weights as defined in (4.3), we compute a maximum weight matching of the tips which correspond to

microtubule tracks. From graph theory, we know that a *vertex disjoint path cover* C is a covering of G where each vertex of G is in some path of C and each vertex belongs to one path only [100]. The weight of a path cover is the sum of weights of edges on the path cover. Given an initial graph G, the problem of finding the best microtubule tracks corresponds to finding the *maximum weight path cover* of G with the weights defined by the similarity in (4.3). Formally, a maximum weight path cover $C(G)$ is a path cover which satisfies:

$$C(G) = \arg\max_{C_i} W(C_i) \tag{4.4}$$

where $W(C_i) = \sum_{(u,v)\epsilon C_i} Sim(u,v)$ and u, v are two vertices in G for which the similarity is computed as in (4.3).

We proceed to compute the maximum weight path cover as suggested in [100]. Let us define a split graph G_{split} corresponding to G as a bipartite graph with partite vertix sets V_+ and V_-. V_+ and V_- are copies of the vertices V. An edge e_{uv} between two vertices u and v in G has the same weight as the edge $e_{u_+v_-}$ in G_{split}. The edges of maximum matching of the bipartite split graph G_{split} correspond to the edges of maximum path cover of G. The algorithm used to compute the maximum matching is depicted in Alg. 1. The graph edge weight computation is detailed in Alg. 2 and Alg. 3.

Algorithm 1 Tracks = ComputeMTtrackingTips(Vid,T,$DistTh$,$MissedFrames$)

Input: Filtered Microtubule video stack Vid having T frames, Euclidean distance threshold for tip similarity $DistTh$, maximum allowed number of frames missed in a MT track $MissedFrames$

Output: Set of microtubule tip tracks $Tracks$

/*Detect tips in all the frames of Video Vid $Tips$ is an array. Each element in $Tips$ has the frame and tip number as contents. */
Tips = DetectTips(Vid);

/*Construct graph $G(V, E)$ with the vertices as $Tips$ and edges E as the similarities between the tips */
E = ComputeWeights(G,Vid,Tips,DistTh,MissedFrames);

/*Compute split graph G_{split} for G */
$V_+ = V$;

$V_- = V$;

for *all edges* E_{uv} *in* G **do**

  $E_{split}(u_+, v_-) = E_{uv}$;

end

$G_{split} = (V_+, V_-, E_{split})$;

/*Do Bipartite Graph matching on G_{split} */
MatchedEdges = BipartiteMatching(G_{split});

/*Stitch together the tips forming an MT track */
Tracks = StitchTips(MatchedEdges);

Algorithm 2 EdgeWeights = ComputeWeights(G,Vid,Tips,DistTh,MissedFrames)

Input: Graph G, Vid, Euclidean distance threshold for tip similarity $DistTh$, maximum allowed number of frames missed in a MT track $MissedFrames$

Output: $EdgeWeights$ of graph G

for $i \leftarrow 1$ ***to*** $|Tips|$ **do**

  for $j \leftarrow 1$ ***to*** $|Tips|$ **do**

    /*$frame(i)$ is the frame number for $Tips(i)$ */

    /*Check if the frames of the tips are within the allowed frame difference and the Euclidean distance between tips is $\leq$ $DistTh$ */

    if ***(***$0 < frame(j) - frame(i) \leq MissedFrames$ & $EucDist(Tips(i), Tips(j)) \leq DistTh$***)*** **then**

      /*$Vid(i)$ is the i^{th} frame in Vid */

      EdgeWeights(i,j)=ComputeSimilarity(Tips(i),Tips(j),Vid(i),Vid(j)));

    else

      EdgeWeights(i,j) = 0;

    end

  end

end

Algorithm 3 Similarity = ComputeSimilarity($Tip_1, Tip_2, Frame_1, Frame_2$)

Input: $Tip_1, Tip_2, Frame_1, Frame_2$

Output: $Similarity$

/*Project Tip_1 on $Frame_2$ */

$d_{geod_1}(Tip_1, Tip_2) = GeodesicDistance(Tip_1, Tip_2, Frame_2)$;

/*Project Tip_2 on $Frame_1$ */

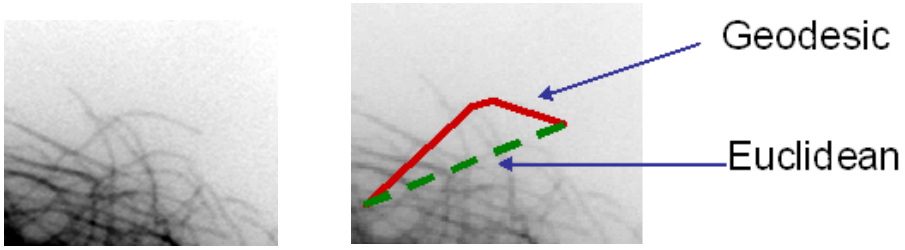
$d_{geod_2}(Tip_1, Tip_2) = GeodesicDistance(Tip_1, Tip_2, Frame_1)$;

$Similarity = e^{-min(d_{geod_1}, d_{geod_2})}$;

4.3 Microtubule Body Formation Based on Geodesics

After computing the maximum weight match for the constructed graph of tips, we can use the microtubule tracks formed by the matched tips to compute dynamic parameters of importance of microtubules such as growth and shortening rates. However, computing microtubule dynamic parameters from the tip locations results in inaccurate lengths when the microtubule shape cannot be approximated by a straight line. Moreover, microtubules move laterally or change shape from frame to frame which affect the calculated length in the case of following tips. For instance, manual methods compute the lengths in frames by taking the Euclidean distance between tip location and an arbitrarily selected reference point on microtubule body, which are then used to find the growth or shortening values in consequent frames. Instead, computing the length as the shortest path along the microtubule body (i.e. a geodesic) is a more accurate alternative (see Fig.4.5). Beside accurate length computation, tracking the microtubule body enables further insights such as studying the effect of different experimental subjects on the curvature of the microtubule, which was not possible before.

The microtubule body in the first frame f_k of the track is computed as follows. Denoting the tip location on the microtubule of interest as t_{start}, the goal is to find a point t_{end} on the microtubule to form the body. For this purpose, let us first define the set P of points satisfying:



(a) (b)

Figure 4.5: Estimate of length (a) original, (b) Euclidean vs geodesic.

$$P = \{t_i \forall (\int_{t_{start}}^{t_i} I^f(s)ds) < \xi\} \tag{4.5}$$

where $I^f(.)$ is the second derivative of Gaussian filtering of the image frame $I(.)$:

$$I^f(x,y) = \max_{\theta}(I(x,y) * G''_{\sigma,\theta}(x,y))$$

In other words, the set of points P is the one for which the geodesic distance from the tip t_{start} is below the threshold ξ determined experimentally. Using the set P, t_{end} is defined as maximizing the following:

$$t_{end} = \arg\max_{cand \in P} \|t_{start} - t_{cand}\|^2 \tag{4.6}$$

This is equivalent of finding the path with minimum curvature originating from the tip. A gradient descent is used to trace the microtubule body from the tip to the ending point determined on the microtubule. The procedure of tracing the microtubule body is depicted in Fig. 4.6. To compute the set of points P which have the geodesic distance

from the tip t_{start} below the threshold ξ, we start by computing a geodesic distance transform with the initial starting point at the tip.

By computing t_{end}, the microtubule body is now represented by two end points, one is the tip and the other one we call the *fixed point*. The name *fixed point* comes from the fact that it is kept fixed along the microtubule track for the remaining frames. For subsequent frames, the problem of extracting the body of the microtubule becomes microtubule body deformation between the fixed point and the tracked tip in every frame. The deformation of the microtubule body is carried out using an active contour based on line features and curve smoothness constraints explained in the next section.

4.3.1 Active Contour for Microtubule Body Tracking

"Snakes" are deformable contours that are initialized on the image plane and allowed to evolve under the influence of a set of internal and external forces. Let the contour be represented parametrically as $C(s)$ where $s \, \epsilon \, [0, 1]$, an energy functional that needs to be minimized is defined as follows:

$$E(C(s)) = \int_0^1 (E_{\text{int}}(C(s)) + E_{ext}(C(s)))\, ds \tag{4.7}$$

where $E_{\text{int}}(C(s))$ is the internal snake force composed of a balanced weight of tension and rigidity of the contour. Since microtubules appear as curvilinear structures in an image $I(x, y)$, *ridge* (accordingly *valleys*) features are used as the external force $E_{ext}(C(s))$. Ridge features can be detected using a second order derivative of a Gaus-

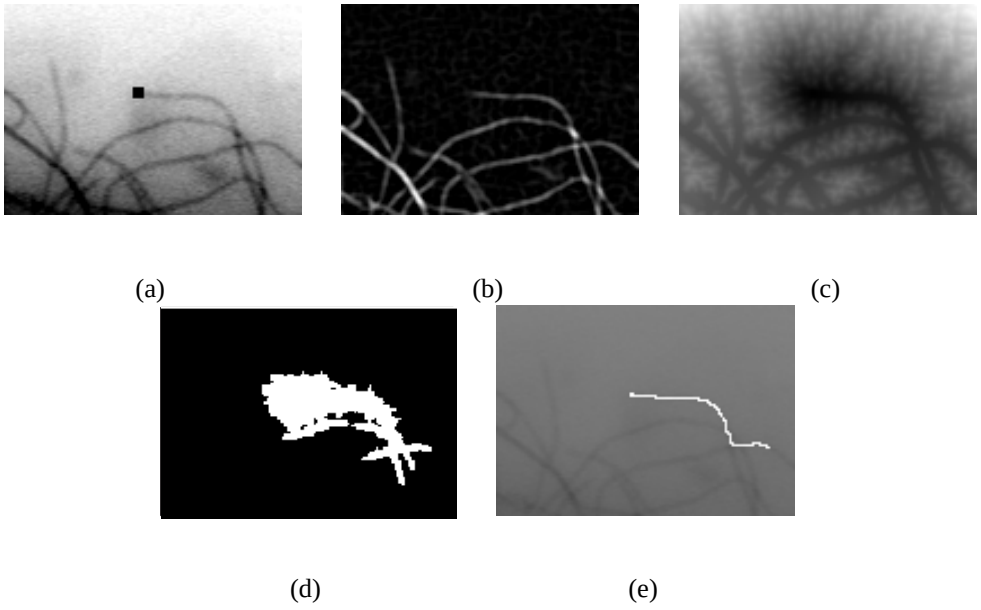


Figure 4.6: Microtubule body formation in the first frame of a track (a) a window around the considered tip in the track with tip overlaid as a black square on the window, (b) the filtering result of the window used as an input to the geodesic distance transform, (c) the distance transform from the tip with darker values denoting smaller distances, (d) points satisfying a distance threshold less than 1, (e) the extracted microtubule body.

sian $G''_\sigma(x, y)$ with a scale σ. Consider the following external force:

$$\nabla E_{ext} = w_1 \left(-\nabla L\right) + w_2 \, L \, sgn(\langle -\nabla L, \overrightarrow{N} \rangle) \, \overrightarrow{N} \tag{4.8}$$

where the first term

$$L(x, y) = \frac{1}{1 + |G''_\sigma(x, y) * I(x, y)|^2}$$

is a gradient vector field created from the line detector response $|G''_\sigma(x, y) * I(x, y)|$. The purpose of this vector field is to pull the active contour towards the desired curvilinear structure of the microtubule. The second term $L.sgn(\langle -\nabla L, \overrightarrow{N} \rangle).\overrightarrow{N}$ is a balloon-based term used to speed convergence of the contour and to help moving the contour in smooth areas. The sign in $L.sgn(\langle -\nabla L, \overrightarrow{N} \rangle).\overrightarrow{N}$ is inspired by the work of [69]. It is clear from (4.8) the connection with geodesic active contours evolution equation [17], but here the ridge features are used instead of the edge features.

The classical way of solving the minimization of the function in (4.7) is to use the Euler-Lagrange condition based on calculus of variations principles. The Euler-Lagrange condition for (4.7) is:

$$-(\alpha C'(s))' + (\beta C''(s))'' + \nabla E_{ext} = 0 \tag{4.9}$$

$C'(s)$ and $C''(s)$ are the tension and rigidity of the contour, respectively. $C'(s)$ and $C''(s)$ are computed as the first and second order derivative of the contour, respectively. All derivatives in (4.9) are computed with respect to the curve parameter s. In the case

of constant weights α and β on the contour and a 2-D active contour problem (on the (x,y plane)), the Euler-Lagrange in (4.9) gives the two conditions:

$$-\alpha x''(s) + \beta x''''(s) + \frac{\partial E_{ext}}{\partial x} = 0 \tag{4.10}$$

and

$$-\alpha y''(s) + \beta y''''(s) + \frac{\partial E_{ext}}{\partial y} = 0 \tag{4.11}$$

In the discrete image space, the contour derivatives are approximated using finite differences leading to the following pair of equations:

$$\begin{aligned} Ax + \frac{\partial E_{ext}}{\partial x} &= 0 \\ Ay + \frac{\partial E_{ext}}{\partial y} &= 0 \end{aligned} \tag{4.12}$$

where A is a penta-diagonal matrix of snake parameters expressed in terms of the weights α and β, as shown in Fig. 4.7.

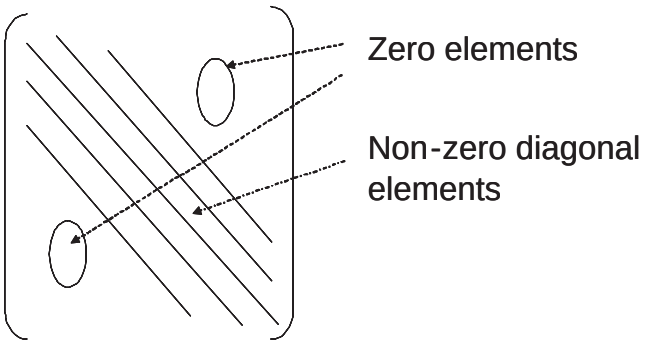


Figure 4.7: An illustration of the penta-diagonal structure of the matrix A in (4.12).

It is worth noting here that at end points of the contour, we need to add external constraints when computing finite differences. For example, in case of a closed contour, circular shifting is usually used. In the case of an open ended contour, linear extrapolation at end points can be employed. Solving for (4.12) implies searching for all possible contours on the digital image space to attain the balance of external and internal forces. Since this is a very expensive solution, we resort to local methods by assuming an initial contour location near features of interest, and allow the contour to evolve using ideas from curve evolution by introducing an artificial time parameter t:

$$C^t = (I + \tau A)^{-1} C^{t-1} - \tau \nabla E_{ext}(C^{t-1}) \tag{4.13}$$

where I is an $N \times N$ unit matrix for a contour represented using N points in the digital space, C^t and C^{t-1} are the contours representations in two successive evolution steps and τ is the time step used in the discretization.

4.3.2 Tracking Results and Quantitative Performance Evaluation

We have applied our automated tracking algorithm on 250 microtubule videos, generating on average 20-25 microtubule tracks per video. The microtubule videos used for tracking were supplied by Kathy Kamath from the Neuroscience Research Institute at University of California Santa Barbara [4]. The videos are based on two microtubule studies of the effect of: a) different drugs, and b) the isotype β-III on microtubule dynamics. In our implementation of the spatiotemporal graph matching, we allowed up to three missing frames between tips of the same microtubule track. For the computation

of the geodesics, we used the Fast Marching algorithm [99]. Consider the computation of a geodesic distance $T(x, y)$ on a 2-D weighted flat domain from an initial point (x_1, y_1). This results in solving the following 2-D eikonal equation:

$$|\nabla T(x, y)| = F(x, y) \tag{4.14}$$

with the boundary condition that $T(x_1, y_1)$ = 0. The 2-D equation is numerically approximated by using a upwind scheme for the computation of partial derivatives [70]:

$$(max\{T_{ij} - min\{T_{i-1,j}, T_{i+1,j}\}, 0\})^2 + (max\{T_{ij} - min\{T_{i,j-1}, T_{i,j+1}\}, 0\})^2 = (F_{ij})^2 \tag{4.15}$$

where $T_{ij} = T(i\delta x, j\delta y)$ with $(\delta x, \delta y)$ is the discretization step in the 2-D domain taken to be $(1, 1)$ and $F_{ij} = F(i\delta x, j\delta y)$ taken as the filtered image frame. Equation (4.15) is then efficiently solved by following the order of propagation of information from the initial point (x_1, y_1) using a binary heap data structure. The computational complexity of the Fast Marching algorithm is $O(MlogM)$ where M is the number of points in the 2-D plane for which $T(x, y)$ is computed.

The complete tracking of microtubules within a video of 30 frames takes approximately 30 minutes using a Matlab implementation on a 3 GHz P-IV machine with 1G RAM. Fig. 4.8 to Fig. 4.14 show example results on some of the frames. To evaluate the tracking performance of microtubules, we manually tracked microtubule tips in two video sequences. The tip tracking performance is shown in Table 4.1. The average microtubule track duration in these videos is 25 frames. The computed errors are

Microtubule video	# MTs	μ_{error}	σ_{error}
1	10	2.25	2.64
2	16	2.85	4.36

Table 4.1: Microtubule tip tracking performance. The average duration of the microtubule video tracks is 25 frames.

within the acceptable margins for biologists for further manual/computer analysis. As we mentioned before, there are no currently publicly available methods and datasets where such a performance has been documented.

4.4 Summary

We have presented in this chapter a novel, fully automated, tracking technique for microtubules. The technique addresses challenges in microtubule videos such as low signal-to-noise ratio, frequent occlusions, and lateral motion of flexible structures. Tracking is based on a spatio-temporal contour tracking approach that can handle missing features and resolve tracking conflicts. The features used for tracking the microtubules are the tips detected, as discussed in Chapter 3 (Section 3.2.3). The detected tips from all the frames are used to construct a graph. By computing a maximum matching over the graph, we can generate microtubule tracks. Based on the maximum matching results of tips, microtubule are formed using a Fast Marching technique and deformed

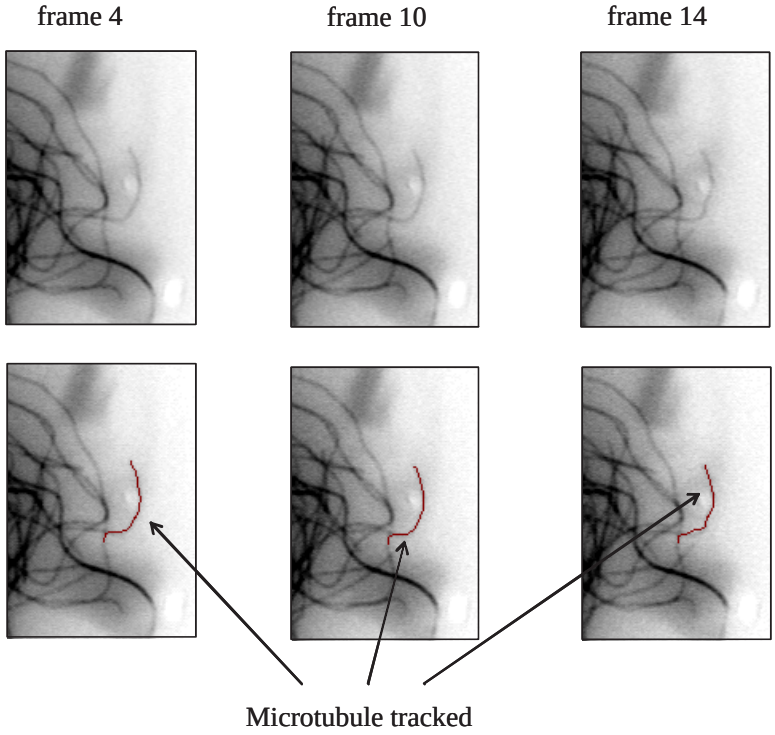


Figure 4.8: Example frames (4, 10, and 14) from automatically computed microtubule tracks.

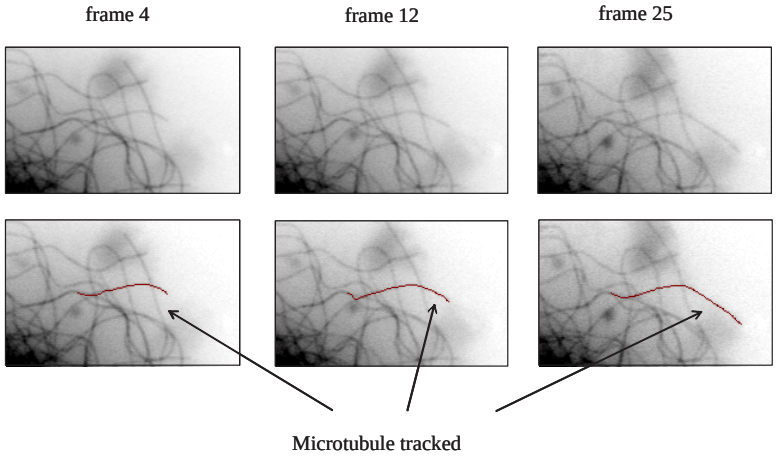


Figure 4.9: Example frames (4, 12, and 25) from automatically computed microtubule tracks.

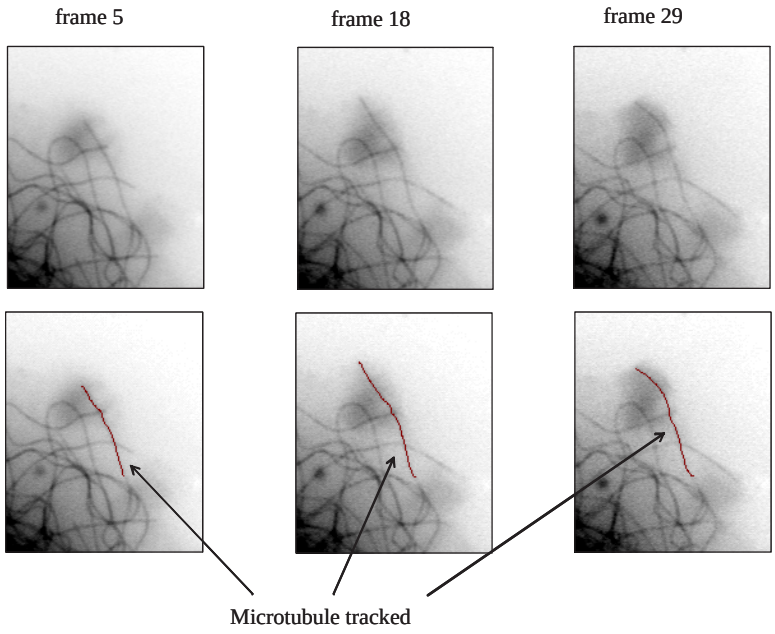


Figure 4.10: Example frames (5, 18, and 29) from automatically computed microtubule tracks.

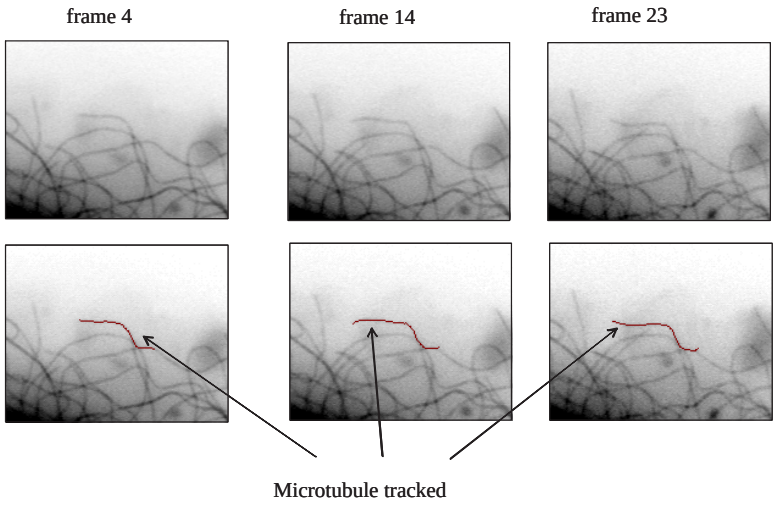


Figure 4.11: Example frames (4, 14, and 23) from automatically computed microtubule tracks.

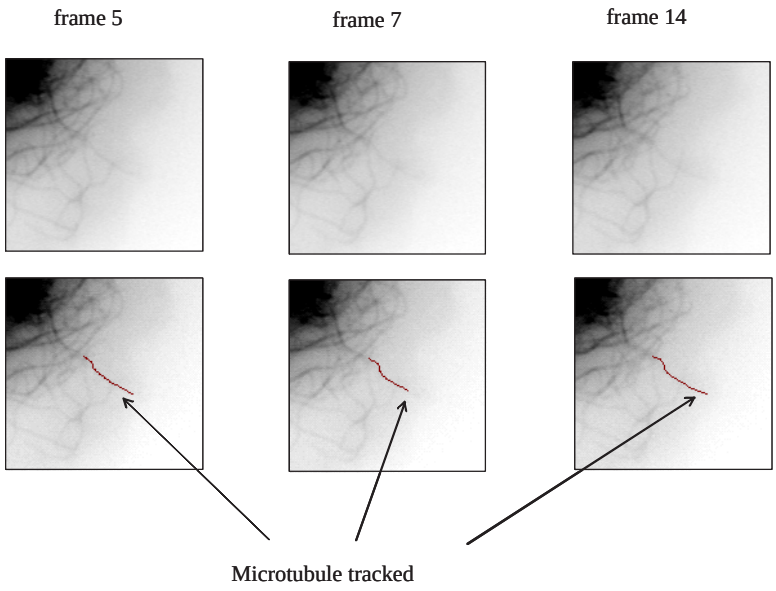


Figure 4.12: Example frames (5, 7, and 14) from automatically computed microtubule tracks.

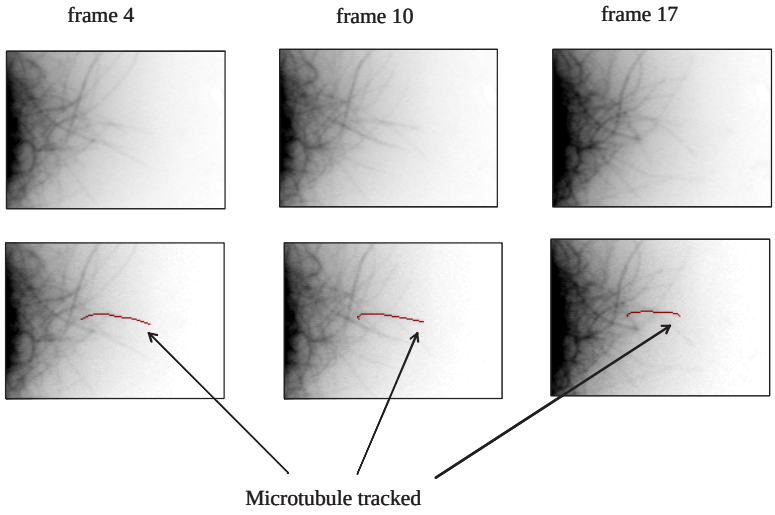


Figure 4.13: Example frames (4, 10, and 17) from automatically computed microtubule tracks.

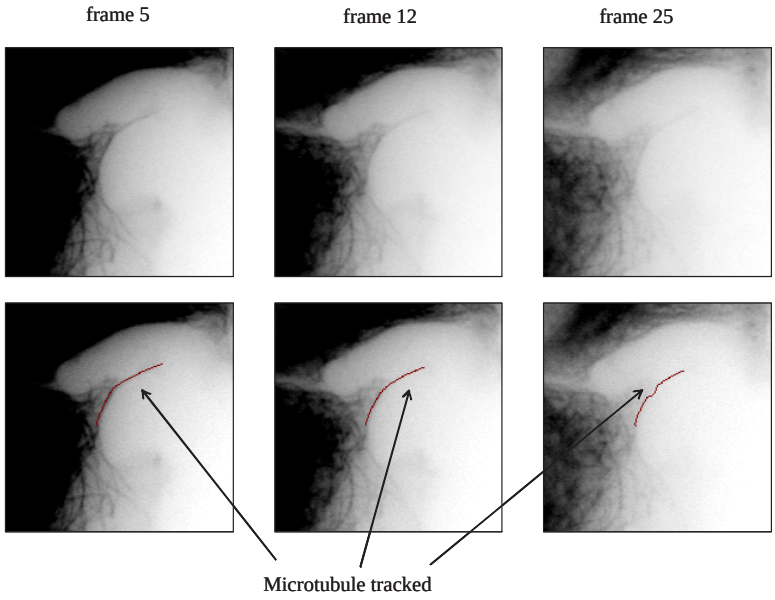


Figure 4.14: Example frames (5, 12, and 25) from automatically computed microtubule tracks.

in every frame using an active contour. We presented quantitative evaluation of the tracking performance for a set of manually tracked microtubules. By generating a large number of full-body microtubule tracks, very useful and previously non-acquirable data can be harvested. Example applications include: 1) quantifying the effect of different experimental conditions on microtubule shape since we have full body tracking, and 2) modeling the full time series of the microtubule tracks using statistical tools for a better understanding of the underlying cell mechanisms regulating microtubule behavior.

Chapter 5

Information Extraction from Microtubule Videos

The tracking of microtubules, whether carried out manually or automatically, produces a time series corresponding to the length of each microtubule under a specific experimental condition. The traditional method of analyzing these time-series data is to calculate statistical parameters characterizing the dynamic behavior of microtubules such as growth and shortening rates. However with the capability of automatically tracking microtubules, we have a rich dataset of full body microtubule tracks on which we can apply powerful data mining tools and statistical models to discover new information previously inaccessible. In this chapter, we start by motivating the approach of using statistical modeling tools for microtubule tracking data. Our first contribution is in investigating the use of hidden Markov models (HMMs) as a powerful analysis tool and

show how it fits as a model for microtubule dynamics. The main motivation behind the use of HMMs is that we consider the growth and shortening of microtubules as observations we are making of an underlying hidden cell process. Furthermore, HMMs allow us to model the full time history for a microtubule rather than the simple statistical measures currently used. The second contribution in this chapter is to propose a temporal association rule mining algorithm for extracting important patterns of dynamics in microtubule tracks. The hypothesis is that microtubule behavior follows certain common patterns that we can describe, setting the stage for subsequent mechanistic biochemical investigations.

To the best of our knowledge, there is no prior work which uses advanced statistical modeling techniques to uncover mechanisms of microtubule dynamics. This is mainly due to the lack of automated feature extraction methods needed for the use of modeling techniques. The main motivation behind exploring statistical modeling tools is that microtubule dynamics are a fundamentally important feature of essentially all eukaryotic cells, yet it is also a process that is poorly understood, having many ill-defined regulating factors influencing it. Statistical tools allow us to fuse a number of different influencing factors to arrive at meaningful inferences. The statistical nature of the modeling process allows a degree of uncertainty which is a characteristic of many biological processes, microtubule dynamics being one of them. For a given experimental condition, we can not guarantee that the behavior of all microtubules within a cell will follow a deterministic and known trend.

The rest of the chapter is as follows. In Section 5.1, we outline the main information extraction tasks capable of advancing microtubule research. We specifically discuss the application of classification, retrieval, modeling and pattern mining techniques for microtubule videos. Applicable related work on general image and video is overviewed in Section 5.2. We motivate the use of statistical modeling tools for microtubule videos in Section 5.3. Specifically, we construct hidden Markov models for microtubule dynamics in Section 5.4. These models are used to quantify similarities in microtubule dynamics between different experimental conditions. Extraction of temporal patterns of dynamics of microtubules is addressed in Section 5.5 using a temporal rule mining algorithm. Finally, we give a summary of the chapter and the main contributions in Section 5.6.

5.1 Information Discovery tasks for Microtubule Videos

We first highlight the potential applications of data mining tools for microtubule videos, and then present a literature review on current approaches to extract information for general classes of video data. Examples of the information discovery tasks that can be carried out on microtubule videos include:

1. *Classification/Clustering of microtubule videos*: The goal here is to discover common characteristics of different effectors, such as drugs or regulatory proteins on microtubule dynamics.

2. *Modeling microtubule dynamics*: By developing models of microtubule dynamics under different experimental conditions, one can acquire a better understanding of the underlying molecular mechanisms governing microtubule activity. Furthermore, model scoring can be used as a similarity measure between different experimental conditions as well as a prediction for new treatments.

3. *Similarity search*: Given a database of microtubule videos or microtubule tracking data, one can search for similar microtubule videos in terms of dynamics or other attributes such as microtubule contour curvature. A possibility here is to use model scoring as a similarity measure between different videos.

4. *Identifying common patterns of growth and shortening in a given experimental condition*: An important biological question is to define different experimental conditions that yield similar consequences. For example, we can quantify how likely a large growth event is followed by an attenuation for cells in various experimental conditions. To evaluate such temporal patterns, one can custom design association rule mining algorithms to compute frequent patterns in the microtubule tracking data.

In the literature, very few papers actually deal with information extraction or modeling of microtubule videos. Example works for the physics and mechanical-based modeling of microtubule data include [45], [46], [79], [80], [84], [91]. In [91], the authors study, via numerical simulation on simplified physical models, how the interaction between microtubules and the free tubulin affect the spatial organization of microtubules.

In [80], the authors use cytoplasmic fragments of melanophores and cytoplasts of BS-C-1 cells to study microtubule centering mechanisms. Centering refers to microtubules arranging themselves in a radial fashion from the center point of a microtubule aster. The nature of the microtubule centering mechanism around the central position aster is still poorly understood. Using live imaging, with mathematical modeling and computer simulation, the authors demonstrate that the microtubule aster discovers a central location in the cytoplasm by moving along spontaneously nucleated non-astral microtubules towards a point at which microtubule nucleation events occur equally on all sides.

Statistical techniques have received less attention from researchers for microtubule dynamics studies [33], [74], [115]. However, we are not aware of any prior work trying to develop generative statistical models for microtubule dynamics under different experimental conditions. In [115], the authors propose a novel algorithm based on multiscale trend analysis for the decomposition of organelle trajectories, driven by microtubule motors, into plus-end or minus-end runs, and pauses. This algorithm is self-adapted to the characteristic durations and velocities of runs, and allows reliable separation of pauses from runs. In [33], the dynamics of the kinetochore microtubules in G1-phase yeast cells are automatically quantified. Two tags are automatically tracked to quantify the dynamics: one close to centromere (CEN) and the other fused to spindle pole (SPB). The dynamics are statistically classified, through hypothesis testing, into the three main types: poleward, anti-poleward and pause. In [74], the transport of pigment granules by

motors along microtubules is studied. Automated tracking of pigment granules along microtubules is carried out. The computed granules velocities have a multimodal distribution. Hence, the authors in [74] propose a model in which multiple motor molecules are responsible for granule transport across microtubules.

Based on the discussed areas where information discovery can be very useful for microtubule videos, we review related work in multimedia mining, for general video and image data, in four main areas:

- Classification/clustering of video data.
- Content-based video retrieval and search.
- Extracting multimedia association rules.
- Modeling activity and video content.

5.2 Related Work on Information Discovery in Image and Video Data

5.2.1 Classification/Clustering of Video Content

Classifying/clustering video content has been a very active research area. Examples include [86] where the authors introduce the concept of a *multiject* (probabilistic multimedia object) that represents several high level concepts occuring in the video. Multijects can be of three types : a) objects (cars and trees), b) sites (indoor, beach), c) events

(explosion, basket scoring). Classification is performed through the use of Bayesian Belief Networks (BBNs). In [117], tree classifiers are used to classify basketball videos into different types of plays based on extracted motion and edge features. In [87], classification is performed on sports videos to separate the two teams' players using a spatio-temporal slice approach by considering the video as a 3-D spatio-temporal volume. A multistage decision system is used in [58] to classify video sports data using characteristic visual cues for each type of sports. The first stage is a decision tree which generate hypotheses of the video content. A hidden Markov model is then used as a second stage to bridge the gap between the user high level concept and the actual visual content. In [78], self-organizing maps are used in generating a texture thesaurus for aerial images based on homogeneous texture features. In [64], a view-based approach is used to recognize humans from their gait. Two sets of features are used for gait description: a) the width of the outer contour of the binary silhouette and the entire binary silhouette. The features are used to train hidden Markov models for the recognition of humans by gait. Hidden Markov models are also used in [103] to recognize different activities performed by humans such as standing up, sitting down and walking. For molecular biological applications, clustering of gene expression data has been previously addressed in the literature [47] and [24] with the motivation that co-expressed genes can possibly reveal common biological function. In [47], a modified version of fuzzy k-means is used to identify groups of functionally related genes. Similarities in gene-expression patterns are used in identifying correlations between yeast genes and

the experimental conditions. Co-expressed gene cluster identification is also addressed in [24] using a density-based clustering.

5.2.2 Extracting Video Association Rules

Extracting association rules in the form of frequent patterns occurring in the data has been extensively studied in the literature for applications such as market basket analysis. There exists in the literature efficient algorithms to extract rules from transactional data [6], [12]. Previous research efforts in multimedia association rule mining include finding spatial co-occurrences of different geographical areas in aerial images [106], where the spatial event cube (SEC) is used to discover elements of first and higher order item sets. Besides, the SEC provides a visualization tool for the different image classes. In [10], authors propose mining techniques such as finding periodic patterns to mine for video editing rules. The extracted rules about video editing could be used in editing another video.

In ClassMiner [118] and more recently in [119], a conventional decomposition of medical video data into shots is used, followed by extracting audio-visual features, and mining three types of events: presentation, dialog, and clinical operation. Mining in this work refers to the classification of the video content into different types of scenes. Based on the video content clustering, a scalable video skimming system is developed which allows access to the video content at various levels of details.

In MultiMediaMiner [114], a hierarchical representation of video assets is devel-

oped based on content or keyword concepts. The system allows similarity comparisons using color of video sequences, and discovers association rules between pairs of feature descriptors corresponding to these sequences. Another example of using multimedia data mining is in generating a good training set for a neural network from a large amount of multimedia data [82] and in automatic feature selection for unsupervised video structure discovery [111]. In [71], the concept of *VideoTrails* is introduced. Video frames are reduced to a trail of points in a low dimensional space. In this space, the frames are clustered, and transitions between clusters are analyzed to characterize the resulting trail. By classifying portions of the trail as either stationary or transitional, gradual edits between shots can be detected. However, the authors in [71] did not address the problem of extracting temporal association rules from the video data.

High-level representation of the structure of the video content was also used in previous work to extract rules about the video content. Examples are in the case of news videos [35], documentaries [77], in sports [116] and in movies [44]. In other approaches, specific domain constraints were used to model certain types of events such as for Basketball videos [98], [117], soccer [72], [49] and tennis video [83], [102].

5.2.3 Content-based Video Retrieval and Search

To automate search by content, different features are extracted from videos, and retrieval is performed using a similarity metric on the extracted features to obtain the most similar objects/regions. In general, video retrieval systems can be broadly classi-

fied into two main types:

- *Object-based systems* [102], [83], [86], [20], [31]: These systems perform first a spatio-temporal segmentation on the video sequence into a set of regions/objects. The queries are issued for specific objects with certain attributes such as color, texture and motion.

- *Holistic systems* [98], [44], [35], [57]: These systems do not perform segmentation, and operate globally on the whole video frame.

In the case of microtubule videos, we are usually interested in estimating the dynamics on a per microtubule basis. This makes the object-based retrieval systems more suitable for use with microtubule video data. For this purpose, the microtubule tracking data can be employed as features for searching.

5.2.4 Modeling Video Data

Modeling the video content has been proposed in the literature for content understanding, classification and searching applications. Example work include the work by Naphade and Huang [86] where they model the video content using the concept of a *multiject* that represents several high level concepts occurring in the video such as explosion. One type of models that gained much interest in the literature of modeling the temporal natural video content as well as recognition of specific activities are the Hidden Markov Models (HMMs) which are doubly embedded stochastic generative

models [92]. In the past, HMMs have been successfully used in numerous applications. Particularly in activity detection and recognition contexts, HMMs were used in human activity recognition [56], abnormal activity detection [109], gesture recognition [15] and American sign language recognition [110]. Another type of models used primarily for time series data -such as microtubule tracking data- are auto regressive moving average models (ARMA) [22] which can be used for classification and prediction. However, the stochastic nature of HMMs make them more flexible and adaptive tools for modeling video content.

5.3 Statistical Modeling of Microtubule Tracks

Traditional measures of microtubule activity consist of simple statistics of dynamics computed from microtubule tracking. The computed statistics combine together different growth and shortening events. Since we do not preserve the entire life history plots, we can not answer questions on common behavior between experimental conditions based on individual tracks. In this chapter, we explore an alternative to the traditional analysis by considering statistical modeling of microtubules.

First, we motivate the use of modeling for microtubules. In general, the goal behind modeling any natural process is to: a) understand how it is behaving and b) being able to predict its outcome in the future. Understanding how a process behaves can help in controlling it, thereby preventing or rectifying any malfunction. Having the ability to predict an outcome of a process can help taking preventive measures if a failure is im-

minent. To relate this to microtubule videos, we aim at building models for microtubule dynamics that give us a better understanding of why microtubules behave as they do for a given experimental conditions. In other words, what are the main factors regulating the microtubule dynamics and how do they work mechanistically? Furthermore, by having models of microtubule dynamics, we can predict behavior thus having a better handle on common behavior under different test subjects such as microtubule associated proteins (MAPs) and drugs. One can suggest new microtubule studies by making predictions based on constructed models for known experimental conditions. One of the main characteristics of microtubule dynamics is that not all the microtubules in one cell show the same dynamic behavior. The variation of dynamic behavior within the same cell, or experimental condition, can be due to many factors, many of which are poorly understood. To account for this uncertainty, we propose the use of statistical modeling tools for microtubule dynamics.

Finally, consider the situation that we want to model a process to understand the underlying factors governing its outcome. We are able to observe its outcome, but we are interested in the underlying unobservable (i.e *hidden*) state of the process that leads to the observed outcome. For example, in the case of microtubule videos, we are observing growth and shortening of microtubules, yet what is more important is to know why microtubules grow or shorten and how they do it. A possibility is that microtubules are growing and shortening to separate the chromosomes in a cell division or to allow transport of proteins from one end of the cell to the other. The molecular mechanism

by which the microtubules grow or shrink is unknown, but it can be modelled as a set of cell states, which are not directly observable. These hidden states can be inferred from a secondary observable output. To statistically model a process where we want to understand the hidden state(s) reflected as observable outputs, we need a doubly embedded stochastic process. This refers to a system whose observable outputs are a stochastic process and whose internal states follow another stochastic process. One of the most powerful and successfully applied doubly embedded stochastic models is the Hidden Markov Model (HMM).

The main features of hidden Markov models that make them a suitable candidate for microtubule dynamics modeling are:

- HMMs are stochastic models that can provide a measure of how well a microtubule behavior matches the trained model. This is very useful for classification, searching and predicting commonalities between different microtubule treatments.

- HMMs allow the emission state observations to be of different lengths, a case that is often encountered with the tracking of microtubules.

- We can interpret the HMM parameters to get a better understanding of the underlying microtubule dynamic process.

- By modeling a process through hidden states, HMMs have the power of relating the observations to something that is not directly observable, hence uncovering

underlying cell states (i.e. molecular events) that affect our observations of dynamics.

5.3.1 Hidden Markov Models (HMMs)

Consider a system that can be described to be in one of N different states at any given time. At uniformly spaced time points, the system can undergo a change of state, or remain in the same state with some probability as shown in Fig. 5.1. This is referred to as a discrete time stochastic process. Let us denote the discrete time instants that the system go through to be $t = 1, 2, 3, ..., T$, and its state at any given time by q_t. For a transition between state i and state j at time t, there is an associated probability denoted by $a_{ij}(t)$. In the general case, specifying $a_{ij}(t)$ for all the states, at every time instant, would involve the current state and all the previous states, such that:

$$a_{ij}(t) = P(q_t = j | q_{t-1} = i, q_{t-2} = k, ...) \tag{5.1}$$

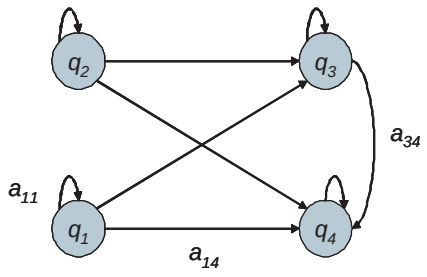


Figure 5.1: A discrete random process with 4 states, q_1 through q_4. The transition probabilities are shown symbolically as $a_{ij}(t) = P(q_t = j | q_{t-1} = i, q_{t-2} = k, ...)$

However, in the case of a first order discrete Markov process, the probability of visiting a particular state depends only on the state at the previous time instant. Hence, $a_{ij}(t)$ is simplified as:

$$a_{ij}(t) = P(q_t = j | q_{t-1} = i) \quad 1 \leq i, j \leq N \tag{5.2}$$

Furthermore, if we assume the system to be time independent, then $a_{ij}(t)$ is not a function of time and can be simply denoted as a_{ij}. To satisfy stochastic constraints, the following condition should hold for a_{ij}:

$$\sum_{j=1}^{N} a_{ij} = 1 \tag{5.3}$$

In the case of a system (or process) whose output is fully specified by the state the system is in, we refer to it as a fully observable process. In other words, knowing the output of the system, we know exactly which state the system is in.

To allow for a greater flexibility in process modeling, we can allow some degree of variability (represented by uncertainty) in the observations at each state. This turns out to be very useful in stochastic system modeling such as in speech modeling systems. In such case the model used to describe the stochastic process will be doubly stochastic in nature. The observation (output) at each state will be random. The underlying stochastic process (the state sequence of the system) is not directly observed, rather indirectly through another stochastic process (the observations). An example of such an embedded doubly stochastic process model is the hidden Markov model (HMM).

Observations and Hidden States

The main elements of an HMM are:

1. *The number of states* N: It is usually an unknown parameter of the modelled process, since these are hidden states. However, depending on the application, one may have a guess on the number of states that the process is likely to go through using prior knowledge of the problem at hand.

2. *The type and number of observations at each state*: There are two different choices for the type of observations at each state, one can use either a discrete set or a continuous set of observations. In the case of a discrete set, we have an M-symbols alphabet observations. Each observation, at a given state, can be one of the possible M symbols from the alphabet $Z = z_1, z_2, z_3, ..., z_M$. In the case of continuous observations at each state, the alphabet size is infinite such as the case when a observation at each state follows a Gaussian distribution.

3. *The initial state distribution*: This is denoted by the π vector whose elements π_i represents the initial state distribution, i.e the probability of the system being modelled by the HMM to start at state i. hence:

$$\pi_i = P(q_1 = i) \quad 1 \leq i \leq N \tag{5.4}$$

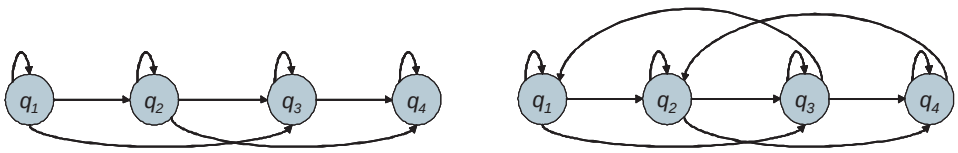
4. *The state transition probabilities*: These are concisely represented by the matrix A whose elements are the a_{ij} with each row satisfying the stochastic constraint in (5.3). The structure of the transition probability matrix is determined by the type of HMM

being used. The two major types are:

1. *Fully connected (or ergodic) HMM*: where all the a_{ij} are allowed to take non-zero values, thus all state transitions are allowed.

2. *Left-to-right (or evolutionary) HMM*: in which case, the HMM is always constrained to start at a specific state and goes, only, forward to another state. In such case there is only a subset of entries in A that are are allowed to assume non-zero values, while other values are clamped to zero.

The two types of HMMs implied by the structure of the transition probability matrix are shown in Fig. 5.2.



(a) (b)

Figure 5.2: (a) An example of a left-to-right HMM, note that the transition are always forward. (b) An example of a fully connected HMM with transitions in both directions.

Example transition matrices for the HMMs shown in Fig. 5.2 are as follows (the matrix on the left is for the left-to-right HMM):

$$\begin{bmatrix} 0.8 & 0.1 & 0.1 & 0 \\ 0 & 0.6 & 0.2 & 0.2 \\ 0 & 0 & 0.2 & 0.8 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad \begin{bmatrix} 0.8 & 0.1 & 0.1 & 0 \\ 0 & 0.3 & 0.3 & 0.4 \\ 0.2 & 0 & 0.5 & 0.3 \\ 0 & 0.5 & 0 & 0.5 \end{bmatrix}$$

Note that for a left-to-right HMM, the last state (N = 4 here) will always have a_N=1.

5. *The observations probabilities at each state*: In a hidden Markov model, we do not know which of the states the system is in. However, we observe at each time instant an output, which we refer to as observation. Depending on the nature of the output observed at each state, we consider the two cases:

1. *Discrete observations*: We denote the observation probabilities by the matrix B with rows denoting the states $1 \leq j \leq N$, and columns denoting each one of the M discrete observation symbols. An element in B denoted by $b_j(k)$ is computed as:

$$b_j(k) = P(o_t = z_k | q_t = j) \tag{5.5}$$

where z_k is the observation symbol at time t and $o_t \in \{z_1, z_2, ...z_M\}$.

2. *Continuous observations*: The continuous observation distribution at each state is described by a probability density function (pdf) $b_j(o)$ at each state j. The form of the probability density function is arbitrary. For example, one of the most

general probability density functions is the mixture of Gaussian distribution at each state:

$$b_j(o) = \sum_{h=1}^{P} \alpha_{jh} N(o, \mu_{jh}, C_{jh}) \tag{5.6}$$

where $N(o, \mu_{jh}, C_{jh})$ is a Gaussian distribution of mean μ_{jh} and covariance C_{jh} in the general case when the observation o is a vector of values. P is the number of Gaussian components in the mixture and α_{jh} is the weight assigned to each Gaussian with $\sum_{h=1}^{P} \alpha_{jh} = 1$

From the above discussion on the HMM elements, we can see that an HMM can be concisely described by its parameters $\lambda = (\pi, \mathbf{A}, \mathbf{B})$, where π_i are prior, $a_{ij} \in \mathbf{A}$ are transition, and $b_{ij} \in \mathbf{B}$ are observation or emission probabilities.

The Main Challenges for HMMs

Assume that we are given a set of L observation sequences $\psi = \{\psi_1, \psi_2, ..., \psi_L\}$ that are emitted from a process we want to model using an HMM. An observation sequence ψ_r in ψ has the form:

$$\psi_r = (o_{r1}, o_{r2}, o_{r3}, ..., o_{rT_r}) \tag{5.7}$$

Note that the time duration of the r^{th} observation sequence is T_r i.e, it can be varied from observation to the other. One of the main advantages of HMM is that it can

handle varying duration observation sequences. To be able to apply HMM for modeling practical stochastic processes, one needs to be able to perform the following tasks:

- Estimate the parameters λ of an HMM given a set of training observation sequences ψ_r for $1 \leq r \leq L$. This corresponds to finding the model parameters $\lambda = (\pi, \mathbf{A}, \mathbf{B})$ that maximize $P(\psi|\lambda)$.

- Given an observation sequence $\psi_r = (o_{r1}, o_{r2}, o_{r3}, ..., o_{rT_r})$, and an HMM with parameters $\lambda = (\pi, \mathbf{A}, \mathbf{B})$, how can we efficiently compute the probability of the observation sequence $P(\psi_r|\lambda)$?

- Given an observation sequence $\psi_r = (o_{r1}, o_{r2}, o_{r3}, ..., o_{rT_r})$, and an HMM with parameters $\lambda = (\pi, \mathbf{A}, \mathbf{B})$, how can we estimate the state sequence that "best" explains the observation sequence?

The first problem of estimating the HMM parameters based on training observation sequences is commonly solved, for the case of maximal likelihood training, using the Baum-Welch algorithm [92] (a version of the EM algorithm [30]). An important sub-problem in training the HMMs is that of initialization. Since the EM procedure is essentially a local optimization algorithm for maximum likelihood parameter estimation, we expect the solution obtained to be only a local one. Hence, different initializations may result in different HMMs being estimated. Example of the approaches used for initialization as discussed in [92] are random initialization and segmental K-means. The second problem of estimating the probability of an observation sequence

to be generated from an HMM with parameters λ is solved using either the forward or backward procedures [92]. Finally, the optimal state sequence is solved using the Viterbi algorithm [92].

5.4 Hidden Markov Modeling of Microtubule Dynamics

The dynamics associated with microtubule growing and shortening, under different experimental conditions, can be modeled using an HMM. The parameters of the HMM of one condition are estimated using training data belonging to the same condition. The number of HMM states is determined experimentally using an independent test set. The standard procedure is to try different number of states and see which one will best fit the available observations or leads to the best performance [64], [103]. Since we are usually faced with the problem of estimating models based on a limited amount of observations, one needs to use an independent test set, sometime called a validation set, to see how well the model fits the data. The purpose of a test set to check for model fitting, rather than the same training data used to estimate the model, is to avoid model over-fitting [23]. Hence, the optimal number of states is estimated as the one maximizing the likelihood over an independent test set. The observations at each state of the HMM are the growth and shortening amounts of a microtubule from a video frame to another. In other words, the observations are the growth (or shortening) rates per

frame. For the case of growth, the observation takes a positive scalar value, while it has a negative value in the case of shortening. We assume that the observations at each HMM state follows a single Gaussian distribution. For each experimental condition, we also choose the type of the HMM –i.e ergodic or evolutionary– by maximizing the likelihood on an independent validation set. The HMM parameters for each experimental condition are trained using the maximum likelihood principle by the Baum-Welch algorithm. By examining the distances between the HMMs of different experimental conditions we assess, in a quantitative manner, the similarity between any two experimental conditions. Though similarity between different experimental conditions is already biologically known for a number of experimental conditions, yet there has been no quantitative measure for it.

5.4.1 Experiments and Results

Dataset

A well established biological observation is that paclitaxel suppresses microtubule dynamics [113]. In this work, we use HMMs for describing microtubule dynamics using a dataset of a study on the effect of a particular tubulin isotype –the βIII [38]. It is known that there are at least seven isotypes of the β-subunit of tubulin in humans, namely βI, βII, βIII, βIVa, βIVb, βV and βVI. These isotypes differ at the protein level at the putative binding site for several microtubule associated proteins [38]. The study in [38] hypothesized that high concentrations of β-III confers upon the cell an acquired

resistance to paclitaxel in terms of dynamics suppression. Let us first describe how the βIII-tubulin is over-expressed in a cell. The DNA encoding βIII-tubulin is transfected into the cells having βI-tubulin. The transfection of βIII-tubulin DNA is performed using an inducible promoter. A promoter is basically a DNA fragment attached to the βIII-tubulin DNA, that activates downstream transcription via tetracycline administration/treatment. However, if the promoter is not induced, the DNA of the βIII will be in the cell, yet with no βIII protein being expressed (this condition is referred to as uninduced βIII-tubulin).

In the study [38], there are five experimental conditions for CHO cells:

1. βI-tubulin: This is considered the control experiment for the normal CHO cells. βI-tubulin is the predominant β-isotype comprising approximately 70% of the total β-tubulin pool. The CHO cells additionally express lower levels of βIVb tubulin and βV tubulin. In this control experiment, βIII-tubulin is not expressed. Microtubules exhibit high dynamicity in this experimental condition.

2. βI-tubulin + paclitaxel: In this condition, the normal CHO control cells were incubated with paclitaxel which normally causes dynamics suppression. Hence the dynamics in this condition are observed to be much slower than the βI-tubulin condition.

3. βIII-tubulin: In this condition, βIII-tubulin has been induced in the cells by transfection with the promoter being induced by tetracycline. As expected, the dynamics in this case are similar to the first case of having βI-tubulin.

4. Uninduced βIII-tubulin + paclitaxel: This is another control experiment where the

DNA encoding the βIII-tubulin was transfected into the cells, but without inducing its promoter. The idea here is to see if just the presence of βIII-tubulin DNA alters the dynamics of microtubules. The observations from the study show that this is not the case. The dynamics of the microtubules were actually suppressed by the presence of paclitaxel, despite the presence of the transfected βIII-tubulin DNA. Hence the resistance acquired by the microtubules against paclitaxel is not due to genetic modification.

5. βIII-tubulin + paclitaxel: This is the main focus of the study, and the central finding in this work. By increasing the concentration of βIII-tubulin the cell environment, and it being used for the assembly/disassembly of microtubules, the paclitaxel drug no longer suppresses microtubule dynamics as was the case in the βI-tubulin + paclitaxel condition. Hence, the hypothesis that the acquired cell resistance against paclitaxel effect is associated with increased levels of βIII-tubulin is supported.

To further clarify the five experimental conditions, we show the presence and absence of the proteins, DNA and paclitaxel in the five experimental conditions in Table 5.1. We denote the five conditions βIII-tubulin, βIII-paclitaxel, uninduced βIII-paclitaxel, βI-tubulin, and βI-paclitaxel by $\{\beta III, \beta IIIt, \beta IIItu, \beta I, \beta It\}$ respectively. The presence of a subject is denoted by (+), while its absence is denoted by (-).

HMM Training and Validation

We automatically tracked microtubules from the videos in [38]. A total of 314 tracks were extracted after a visual inspection of the tracking results. Recalling that the five

	βI protein	βIII DNA	βIII protein	paclitaxel
βI	+	-	-	-
βIt	+	-	-	+
βIII	+	+	+	-
$\beta IIIt$	+	+	+	+
$\beta IIItu$	+	+	-	+

Table 5.1: Presence and absence of different agents in the five experimental conditions in the βIII study [38].

conditions βIII-tubulin, βIII-paclitaxel, βIII-paclitaxel uninduced, βI-tubulin, and βI-paclitaxel are denoted by $\{\beta III, \beta IIIt, \beta IIItu, \beta I, \beta It\}$ respectively. We train a separate HMM for each of the classes. We need to specify the number of states and model type (fully connected or left-to-right) for each class. To avoid overfitting with the finite amount of microtubule tracks for each condition, we use the technique of cross-validation for model selection [23]. We experimented with models having 3 to 5 states with both left-to-right and fully-connected topologies. We use a 4-fold cross-validation.

As an example, consider fitting the data with 3 states with left-to-right topology. We partition the microtubule tracks belonging to one experimental condition into 4 separate subsets denoted by $\{T_1, T_2, T_3, T_4\}$. First, we use $\{T_2, T_3, T_4\}$ for training the HMM and T_1 as a validation set for model fitting. We obtain a log likelihood validation score for the validation set T_1, let us denote the score by $ValidScore_1$. Then, we used T_2

for validation and $\{T_1, T_3, T_4\}$ for training the HMM. We obtain a new log likelihood validation score for the validation set T_2, let us denote the score by $ValidScore_2$. We compute the validation score for all the subsets in $\{T_1, T_2, T_3, T_4\}$. The final validation score for the case of assuming 3 states with left-to-right topology will be:

$$Score = \frac{ValidScore_1 + ValidScore_2 + ValidScore_3 + ValidScore_4}{4} \tag{5.8}$$

We repeat the above procedure, of computing an average validation score for each candidate model, by considering all the allowed number of HMM states and network topologies. Finally, the best model is chosen to be the one that leads to the maximum average validation score. This whole procedure is repeated for the five experimental conditions.

Quantifying Similarities Between Experimental Conditions

Microtubule dynamic behavior is known to be affected by many regulators, including microtubule associated proteins (MAPs) and drugs. For example, both the microtubule associated protein tau and the drug paclitaxel inhibit microtubule dynamics. Similarities between different experimental conditions have thus been established for a number of test subjects, yet primarily in a qualitative manner. We use the constructed HMMs to estimate a quantitative dissimilarity measure between the different experimental conditions. We adopt a popular method in comparing two HMMs discussed in [92]. Consider we have two HMMs with parameters λ_1 and λ_2. We have a set of observations belong-

ing to two experimental conditions denoted by $O^{(1)}$ and $O^{(2)}$ respectively:

$$O^{(1)} = \{o_1, o_2, o_3, ...o_{T_1}\}$$
$$O^{(2)} = \{o_1, o_2, o_3, ...o_{T_2}\}$$

where T_1 and T_2 are the number of observations for $O^{(1)}$ and $O^{(2)}$ respectively. The following measure can be used to find the distance (dissimilarity) between λ_1 and λ_2:

$$D(\lambda_1, \lambda_2) = \frac{1}{T_1}\left(\log P(O^{(2)}|\lambda_1) - \log P(O^{(2)}|\lambda_2)\right) \tag{5.9}$$

This dissimilarity measure quantifies how the observations $O^{(2)}$ match λ_1 versus matching its own model λ_2. To make the measure in (5.9) symmetric, the final dissimilarity measure $D_s(\lambda_1, \lambda_2)$ between models λ_1 and λ_2 is defined as follows:

$$D_s(\lambda_1, \lambda_2) = \frac{1}{2}\left[D(\lambda_1, \lambda_2) + D(\lambda_2, \lambda_1)\right] \tag{5.10}$$

We show in Table 5.2 a five by five matrix showing the distances between pairs of HMMs describing the five experimental conditions in the βIII-tubulin isotype study [38]. The smaller the distance, the more similar are the two experimental conditions. This table of similarities of experimental conditions is useful in assessing, in a quantitative manner, the common and distinct effects of different treatments on microtubules. The main diagonal has all zeros since the distance of a model to itself is zero. For example, consider the distances between the βIII model and the other models. Biologically, it is observed that the cells over-expressing βIII-tubulin exhibit normal microtubule dynamics. This is reflected, in a quantitative manner, in the first row of Table 5.2 as the

distance between βIII and βI conditions is smaller than either of the distances between βIII and uninduced βIII + paclitaxel or between βIII and βI + paclitaxel. In other words, the cells over-expressing βIII-tubulin are more similar, in terms of microtubule dynamics, to the cells expressing βI-tubulin than they are to the cells treated with paclitaxel (both for βI and uninduced βIII cells).

Investigating the second row in Table 5.2, we directly conclude from the distance between the βIII + paclitaxel and βIII and the distance between βIII + paclitaxel and βI + paclitaxel that paclitaxel does not have the same suppressing effect on cells over-expressing the βIII-tubulin protein as β-I. By examining all the similarities in Table 5.2, we can see that all the interrelations between the five experimental conditions are preserved with the exception that the first row corresponding to the βIII condition, in which the distance between βIII and βIII + paclitaxel should be smaller than the distance between βIII and βIII uninduced + paclitaxel condition.

An alternative method to visualize the distances between the five experimental conditions is to embed the distance matrix in Table 5.2 into a two dimensional space using multi-dimensional scaling (MDS) [11]. The resulting two dimensional space is shown in Fig. 5.3. Visually, the two groups of dynamics can be separated as shown by the two ellipses on the figure. However, we observe that the βIII + paclitaxel condition is farther from the βIII and βI conditions, which suggests that, though that all three conditions are characterized by high dynamics, they are not all equidistant in terms of behavior.

	βIII	$\beta IIIt$	$\beta IIItu$	βI	βIt
βIII	0	0.068	0.066	0.029	0.065
$\beta IIIt$	0.068	0	0.080	0.064	0.090
$\beta IIItu$	0.066	0.080	0	0.076	0.049
βI	0.029	0.064	0.076	0	0.070
βIt	0.065	0.090	0.049	0.070	0

Table 5.2: Dissimilarities between the HMMs of different experimental conditions in the βIII study [38]. The five different experimental conditions βIII-tubulin, βIII-paclitaxel, uninduced βIII-paclitaxel, βI-tubulin, and βI-paclitaxel are denoted by $\{\beta III, \beta IIIt, \beta IIItu, \beta I, \beta It\}$ respectively.

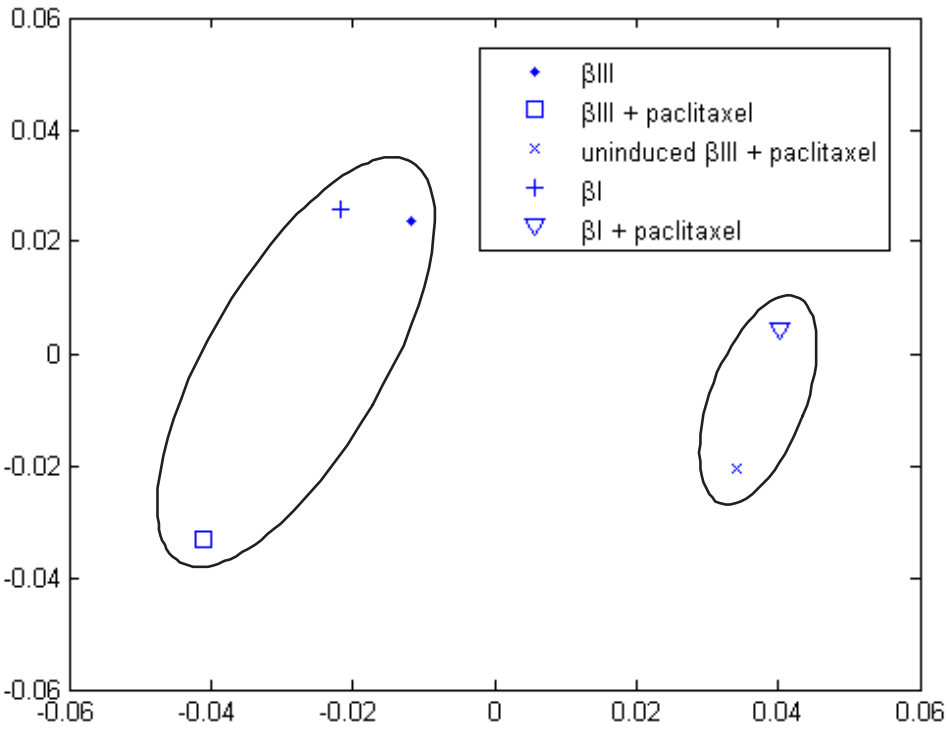


Figure 5.3: A two-dimensional embedding of the distances between the five experimental conditions $\{\beta III, \beta IIIt, \beta I, \beta IIItu, \beta It\}$. The two ellipses represent the two group of dynamics $g_1 = \{\beta III, \beta IIIt, \beta I\}$ and $g_2 = \{\beta IIItu, \beta It\}$.

5.5 Temporal Association Rules in Microtubule Videos

5.5.1 Introduction

Extracting association rules in the form of frequent patterns occurring in the data has been extensively studied in the literature for applications such as market basket analysis. There exists in the literature efficient algorithms to extract rules from transactional data exist in the literature [6], [12]. The premise of using rule mining algorithms for microtubule videos is to extract the frequent patterns in microtubule tracks under different experimental conditions. However, traditional rule mining algorithms do not explicitly address the temporal structure inherent in video data. To that end, we need to represent microtubule videos in suitable forms, from which we can extract frequent rules and patterns of behavior. It is worth pointing here that frequent temporal rule mining should be able to discover rules that involve more than simply two consecutive time points. For example, consider the following events:

- S_1 denotes an average shortening of 6 pixels.
- S_2 denotes an average shortening of 3 pixels.
- S_3 denotes an average growth of 5 pixels.
- S_4 denotes an average growth of 1 pixels.
- S_5 denotes an average shortening of 1 pixel.

Suppose we want to check for an experimental condition, say a paclitaxel treated cell, what are the most frequent patterns of dynamic behavior. A pattern of dynamic behavior is represented as a sequence of events (symbols), such as $S_1 \rightarrow S_2 \rightarrow S_3$. The main requirements to find frequent temporal patterns involving an arbitrary number of events are:

- Representing the microtubule video tracks as symbols in an alphabet corresponding to different amounts of growth and shortening.
- Devising efficient algorithms that can compute arbitrary length frequent patterns.

For the first point, we consider quantizing the growth and shortening amounts in a number of quantization bins and using the bin as a symbol. For the second issue of devising efficient rule mining algorithm, we are presenting a temporal rule mining algorithm. The block diagram in Fig. 5.4 shows the proposed microtubule temporal rule extraction flowchart.

5.5.2 Temporal Association Rule Mining Algorithm

In the following, we introduce a video rule mining method that takes into account the temporal structure of video. The method can detect temporal ordering and corresponding correlations among symbols denoting different amount of growth and shortening in microtubule tracks. In this sense it differs from the standard association rule mining methods [6], [12] that do not explicitly address the temporal structure inherent in such

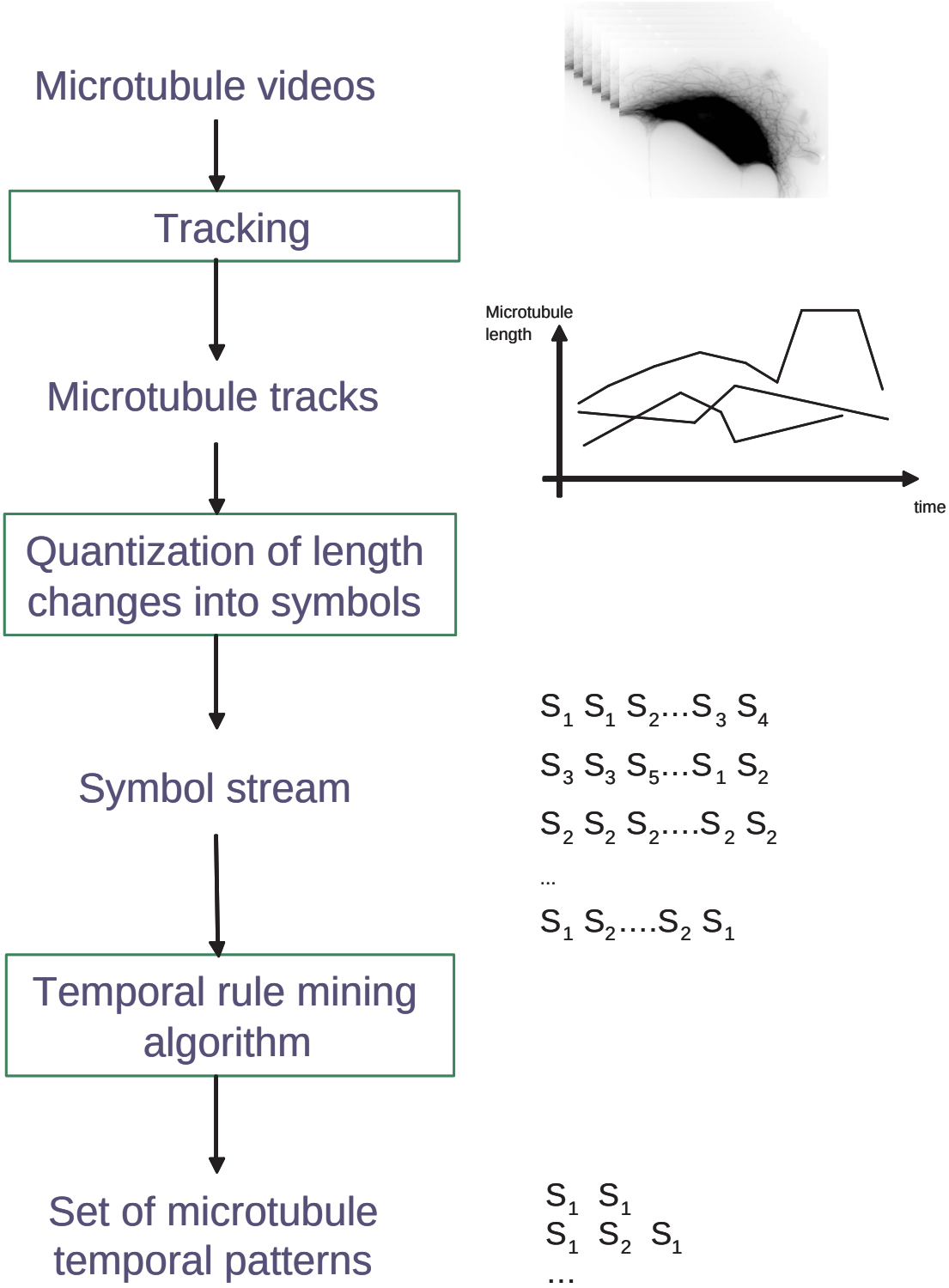


Figure 5.4: The flowchart of the proposed microtubule temporal rule mining algorithm.

data. The proposed method is based on using a multi-level association rules mining approach and borrowing ideas from the well know Apriori algorithm for association rule mining [6]. Let $Level$ be the length of a rule pattern that is discovered. When the level increases, the rule pattern becomes longer and more complex. For instance, the rules at $Level$ 2 can be:

$S_1 \rightarrow S_2$

which means that state S_1 of 6 pixels average shortening is followed by state S_2 of 5 pixels average shortening (the numbers are just used as examples). Another example of a rule at level 3 can be:

$S_3 \rightarrow S_2 \rightarrow S_3$

which means that state S_3 of 5 pixels average growth is followed by state S_2 of 5 pixels average shortening, then finally S_3 of 5 pixels average growth.

To describe how the temporal association rule mining algorithm proceeds, let us first define some terms. For a rule R having the form: $S_1 \rightarrow S_2 \rightarrow ...S_l \rightarrow S_e$, we define its *support* to be the probability of occurrence in the symbol stream:

$$support = P(S_1 \cup S_2 ... \cup S_l \cup S_e), \tag{5.11}$$

If the computed support of a rule is greater than a pre-specified minimum support value, then the rule is referred to as *strong*. The main goal of the temporal association rule mining algorithm proposed here is to extract all the strong rules at all levels. Since checking for every possible candidate rules will be computationally very expensive,

we use the common anti-monotone property of the support measure of a rule in order to prune the search space: *The sub-rule (part of the rule) of a strong rule must also be a strong rule*, i.e. if $S_3 \rightarrow S_2 \rightarrow S_3$ is a strong rule at level 3, both $S_3 \rightarrow S_2$ and $S_2 \rightarrow S_3$ must be strong rules at level 2. This property is used in our multi-level association rule mining algorithm (Algorithm 4) to prune the search space of frequent temporal rules. In this algorithm, the input is a set of R vectors, each of which is a symbol sequence for a corresponding microtubule track observation sequence. The set of vectors is denoted by $G = \{g_1, g_2, ..., g_R\}$. Consider that we have N symbols, $S_1, S_2, ...S_N$ each representing an amount of growth or shortening per frame and a minimum support for any rule denoted as *minsupp*. At the first level of the algorithm, we consider rules which consist of just one symbol. We form a candidate set C_1 of rules composed of all the symbols:

$$C = \{S_1, S_2, ...S_N\}$$

We compute the support of each of the candidate rules in C_1. We delete all the rules that have a support less than *minsupp*. The remaining rules are the strong rules discovered of length-one symbol. In level 2, we first form the candidate set of rules by only considering the strong rules at the previous level (level 1). This is how we prune the searching for rules. Consider for example that the strong rules at level 1 are S_1, S_3 and S_9. The candidate set of rules in level 2 will be formed by trying all different length 2 combinations from the strong rules at level 1. The combinations are order sensitive, so we consider $S_1 \rightarrow S_3$ and $S_3 \rightarrow S_1$ as different. The candidate set of rules al level 2 is:

$C_2 = \{S_1 \rightarrow S_1, S_1 \rightarrow S_3, S_1 \rightarrow S_9, S_3 \rightarrow S_1, S_3 \rightarrow S_3, S_3 \rightarrow S_9, S_9 \rightarrow S_1, S_9 \rightarrow S_3, S_9 \rightarrow S_9\}$

Again, we prune all the rules that have a support less than *minsupp*. The same idea of forming a candidate set of rules and then pruning a subset of them by the *minsupp* value is applied to all higher levels.

5.5.3 Experimental Results and Discussions

The input to the microtubule temporal association rule algorithm is a set of R vectors, each of which is a symbol sequence for a corresponding microtubule track observation sequence. The set of vectors is denoted by $G = \{g_1, g_2, ..., g_R\}$. We extract frequent temporal rules for the microtubule tracks for the two groups $g_1 = \{\beta III, \beta IIIt, \beta I\}$ and $g_2 = \{\beta IIItu, \beta It\}$ respectively. We use a uniform quantizer for the growth and shortening amounts, and use the bin centers as symbols. We will thus have a fixed variance, and we can arbitrary choose the number of bins to have a fine or coarse quantization, and consequently temporal rules. By virtue of a fixed quantization step, a symbol can be directly related to an amount of growth or shortening. We considered the following quantization centers: $\{-12, -6, 0, 6, 12\}$ based on the actual differences in the microtubule tracking data. We have 5 symbols, S_1 through S_5, for example the state S_3 corresponds to the range $[-3\ 3]$. We used the five symbols to quantize the differences in microtubule length from frame to frame using the nearest neighbor rule. The results are shown in Fig. 5.5 for a minimum support of 2%. In these results, the

Algorithm 4 R_{total} = RuleMining $(G, S, N, minsupp, NumLevels)$

Input: Symbol sequence G, symbol alphabet S, number of symbols N, minimum support $minsupp$, number of levels $NumLevels$

Output: Strong temporal rules R_{total}

$R_{total} = \phi$

$R_1 = ComputeRulesFirstLevel(G, S, N, minsupp)$;

$R_{prev} = R_1$;

for $level \leftarrow 2$ ***to*** $NumLevels$ **do**

  $R_{thislevel} = \phi$

  for $j \leftarrow 1$ ***to*** $|R_{prev}|$ **do**

    for $i = 1 \rightarrow |R_1|$ **do**

      $C = [R_1(i)\, R_{prev}(j)]$

      if $Support(C)$ *in* $G >= minsupp$ **then**

        $R_{thislevel} \leftarrow R_{thislevel} \cup C$

      end

      $C = [R_{prev}(j)\, R_1(i)]$

      if $Support(C)$ *in* $G >= minsupp$ **then**

        $R_{thislevel} \leftarrow R_{thislevel} \cup C$

      end

    end

  end

  $R_{prev} = R_{thislevel}$

  $R_{total} = R_{total} \cup R_{thislevel}$

end

Algorithm 5 $R_1 = ComputeRulesFirstLevel(G, S, N, minsupp)$;

Input: Symbol sequence G, symbol alphabet S, number of symbols N, minimum support $minsupp$

Output: First level rules R_1

$R_1 = \phi$

for $i \leftarrow 1$ ***to*** N **do**

  if $Support(S[i])$ *in* $G >= minsupp$ **then**

    $R_1 = R_1 \cup S[i]$

  end

end

right column shows the support of the rule and we extracted rules up to the length of 4 symbols. The main observation from the extracted rules is that there $S_1 \rightarrow S_5$ and $S_5 \rightarrow S_1$ are frequent temporal patterns in the case of cells where there is no paclitaxel effect. This means that there is occurrence, above 2%, to have a transition between large growth to large shortening and vice-versa. This is not the case for the cells affected by paclitaxel. An interesting observation is that, although we have frequent rules involving the transition from large growth to large shortening in g_1, the rules comprising three or four symbols do not show the occurrence of either S_1 or S_5. This means that on a longer time span, the frequent rules are those involving either small growth or small shortening only.

Rule	Support
S_1	0.084
S_2	0.154
S_3	0.511
S_4	0.164
S_5	0.086
$S_3\ S_1$	0.024
$S_5\ S_1$	0.024
$S_1\ S_3$	0.025
$S_1\ S_5$	0.022
$S_3\ S_2$	0.064
$S_4\ S_2$	0.043
$S_2\ S_3$	0.076
$S_2\ S_4$	0.034
$S_3\ S_3$	0.323
$S_4\ S_3$	0.069
$S_5\ S_3$	0.027
$S_3\ S_4$	0.076
$S_3\ S_5$	0.027
$S_4\ S_4$	0.026
$S_3\ S_3\ S_2$	0.039
$S_3\ S_2\ S_3$	0.033
$S_3\ S_4\ S_2$	0.023
$S_4\ S_2\ S_3$	0.022
$S_2\ S_3\ S_3$	0.046
$S_3\ S_3\ S_3$	0.220
$S_4\ S_3\ S_3$	0.043
$S_3\ S_3\ S_4$	0.046
$S_3\ S_4\ S_3$	0.035
$S_3\ S_3\ S_3\ S_2$	0.026
$S_3\ S_3\ S_2\ S_3$	0.020
$S_3\ S_2\ S_3\ S_3$	0.022
$S_2\ S_3\ S_3\ S_3$	0.032
$S_3\ S_3\ S_3\ S_3$	0.156
$S_4\ S_3\ S_3\ S_3$	0.026
$S_3\ S_3\ S_3\ S_4$	0.030
$S_3\ S_4\ S_3\ S_3$	0.024
$S_3\ S_3\ S_4\ S_3$	0.021

(a)

Rule	Support
S_1	0.048
S_2	0.146
S_3	0.597
S_4	0.150
S_5	0.058
$S_3\ S_2$	0.070
$S_4\ S_2$	0.036
$S_2\ S_3$	0.081
$S_2\ S_4$	0.034
$S_3\ S_3$	0.417
$S_4\ S_3$	0.076
$S_3\ S_4$	0.079
$S_4\ S_4$	0.022
$S_3\ S_3\ S_2$	0.043
$S_3\ S_2\ S_3$	0.040
$S_3\ S_4\ S_2$	0.022
$S_2\ S_3\ S_3$	0.055
$S_3\ S_3\ S_3$	0.305
$S_4\ S_3\ S_3$	0.046
$S_3\ S_3\ S_4$	0.054
$S_3\ S_4\ S_3$	0.041
$S_3\ S_3\ S_3\ S_2$	0.032
$S_3\ S_3\ S_2\ S_3$	0.025
$S_3\ S_2\ S_3\ S_3$	0.027
$S_2\ S_3\ S_3\ S_3$	0.041
$S_3\ S_3\ S_3\ S_3$	0.227
$S_4\ S_3\ S_3\ S_3$	0.033
$S_3\ S_3\ S_3\ S_4$	0.036
$S_3\ S_4\ S_3\ S_3$	0.026
$S_3\ S_3\ S_4\ S_3$	0.029

(b)

Figure 5.5: Frequent temporal rules of microtubule dynamics for the case of a) group 1, $g_1 = \{\beta III, \beta IIIt, \beta I\}$ b) group2, $g_2 = \{\beta IIItu, \beta It\}$. Symbols are derived from a fixed quantizers with centers $\{-12, -6, 0, 6, 12\}$. The minimum support used for the rules is 2%

5.6 Summary

We have presented in this chapter modeling and pattern extraction tools for microtubule tracking data which can provide novel insights into molecular mechanisms underlying the regulation of microtubule dynamics. Specifically, we have investigated the use of statistical modeling tools and proposed a temporal pattern discovery algorithm. As a powerful statistical modeling tool, we have modeled microtubule tracks using HMMs. In addition to agreeing with the manually established findings on microtubule dynamics, the constructed models provide deeper insights into the mechanisms by which dynamics occur. The HMM states represent the main modes of growth and shortening in the microtubule tracks, while the transitions between states provide estimates on the most frequent dynamic patterns of behavior. We use the constructed models of microtubule behavior for quantifying similarities between different experimental conditions. We can now look at the full life history plots and infer from them specific patterns of dynamics by using pattern mining techniques. The extracted rules show the most frequent patterns of dynamics of microtubules under a specific experimental condition.

Chapter 6

Summary and Future Research Directions

In this dissertation, we have investigated the use of automated tools to facilitate and enhance the study of microtubule videos. The analysis of microtubule dynamics has been mostly manual and limited to simple statistical measures of dynamic parameters. On the data generation side, we proposed automated microtubule detection and tracking techniques to speed up, as well as enhance the quality of microtubule tracking data gathering. In Chapter 3, we proposed an automated technique to extract a band around the cell periphery, the microtubule layer, through temporal clustering in the frequency domain. We devised an automated approach to detect microtubules in this layer using a filtering-based approach to detect the microtubule polymer mass. Microtubule tips are then estimated and used for automated tracking using a graph-based approach (Chapter

4). By solving the tracking problem for all microtubules over the full video time extent, we were able to resolve tracking conflicts and to overcome microtubule tip occlusion and mis-detection. The technique performs well under noisy and illumination varying conditions and is reasonably fast.

On the data analysis side, we investigated the use of advanced statistical modeling and information discovery tools as powerful techniques not previously accessible to biological studies on microtubules (Chapter 5). We investigated the use of of hidden Markov models (HMMs) as rich statistical modeling tools to build prototype models for different experimental conditions. Besides, the built models provide a quantitative measure of similarity between effects of different test subjects, such as MAPs and drugs, on microtubule dynamics. Furthermore, we devised a temporal rule mining algorithm to detect frequent temporal patterns of dynamics for different experimental conditions providing valuable information to biologists previously unattainable. In the following, we highlight some of the potential future research directions for the automated analysis of microtubule dynamics.

6.1 Future Research Directions

6.1.1 Spatial Localization Effects on Microtubule Dynamics

Microtubules do not exist in isolation in cells, and in most cases are surrounded by other microtubules, as well as as other cellular components. Microtubules depend on

the available reservoir of soluble tubulin in the cell environment for their dynamics. It is reasonable to expect that, at an equilibrium stage, if microtubules within a cell region grow, then this needs to be compensated for by others shortening. This is what we refer to as *neighboring effect*. A valuable question to be answered is how to quantify this neighboring effect. It is far from practical to achieve this manually, since a fair quantification of neighboring effect would necessitate the tracking of all resolvable microtubules, a time prohibitive task. However, using the automated tracking developed in this thesis, one can estimate the effect of neighbors for various experimental conditions.

Another interesting question is to estimate the effect of the distance of the microtubule tip from the cell center on the microtubule dynamics. Again, it is reasonable to expect that microtubules can behave differently depending on their positions in the cell, so as to achieve different cellular functions. This is very relevant especially in neuronal growth cones studies.

6.1.2 Studying the Effect of Different Experimental Conditions on Microtubule Curviness

It is has been observed that microtubule body curviness varies significantly from one video to the other and within the same video. An interesting question would be to measure to what extent is the curviness of a microtubule affected by the spatial location within the cell, by the application of a drug or by the presence of a microtubule

associated protein with a specific concentration and mutation. In this context, neuronal growth cones microtubules do exhibit shape deformations of microtubule bundles to direct the growth pattern.

6.1.3 Studying Correlations Between Effects of Drugs and Microtubule Associated Proteins

Microtubule dynamics have been shown to be affected by the presence and concentration of different drugs and microtubule associated proteins. A potential useful investigation can try to find correlations, if any, existing between say paclitaxel and MAP1B. The developed statistical tools for modeling and pattern analysis of microtubule dynamics can be extended to study these correlations. An outcome of such analysis can direct future biological experiments by predicting combined effects of drugs and proteins on microtubule dynamics.

Glossary

Aster

An aster is a radial arrangement of microtubules forming a star shape.

BS-C-1

BS-C-1 are an example of non-pigment cells.

CHO cells

Chinese Hamster Ovary cells.

Cytoplasm

Cytoplasm refers to all the cell components except the nucleus and plasma membrane.

Dimer

A dimer is composed of two monomers [1].

Eukaryotic cells

In Eukaryotic cells, the DNA (deoxyribonucleic acid) is arranged into chromosomes contained within the nucleus [14]

Green Fluorescent Protein (GFP)

This is a fluorescent protein that is coupled to a target protein to serve as a labelling technique (GFP fusion). The basic idea behind fluorescent protein labelling is to genetically or chemically modify a target protein so that it is fluorescent and thus emits light. The light in the case of GFP is green and can be imaged when excited with a light source of the appropriate wavelength.

Half maximal arrest

At half maximal arrest, half of the number of dividing cells could not proceed into mitosis and are blocked.

Melanophores

Melanophores are an example of pigment cells.

Monomer

A monomer is a molecule that can join with other molecules to form a large assembly called a polymer. A monomer is the smallest repeating unit in a polymer chain [2].

Motor proteins

Motor proteins are molecules capable of converting chemical energy into mechanical energy allowing them to move along cytoskeleton components such as

microtubules. This movement facilitates cellular function as vesicle transport and chromosome separation.

Prokaryotic cells

In prokaryotic cells, the DNA is not separated from the rest of the cell in a nucleus [14].

Promoter

A promoter is a DNA fragment attached to a target DNA that activates downstream transcription.

Protein isotypes

Protein isotypes are different forms of the same protein. For example, β-tubulin protein isotypes differ primarily in a protein region that lies on the exterior of the microtubule [38].

Tetracycline

A small antibiotic molecule that is used to induce a promoter to activate DNA transcription into RNA.

Transcription

The process by which DNA is transformed into RNA.

Transfection

Transfection is the process by which a foreign DNA is introduced into a mammalian cell [3].

Translation

The process by which RNA is transformed into protein.

Bibliography

[1] "http://matse1.mse.uiuc.edu/ tw/polymers/glos.html."

[2] "http://www.deh.gov.au/settlements/waste/degradables/glossary.html."

[3] "http://www.iscid.org/encyclopedia/transfection."

[4] *Neuroscience Research Institute, University of California Santa Barbara.*

[5] D. Adalsteinsson and J. Sethian, "A fast level set method for propagating interfaces," *Journal of Computational Physics*, vol. 118, no. 2, pp. 269–277, 1995.

[6] R. Agrawal and R. Srikant, "Fast algorithms for mining association rules," in *Proc. 20th Int. Conf. Very Large Data Bases, VLDB*. Morgan Kaufmann, 1994, pp. 487–499.

[7] O. Al-Kofahi, R. J. Radke, B. Roysam, and G. Banker, "Automated semantic analysis of changes in image sequences of neurons in culture," November 2005.

[8] B. Alberts, D. Bray, J. Lewis, M. Raff, K. Roberts, and J. D. Watson, *Molecular Biology of the Cell*, 4th ed. Garland Science, 2002.

[9] A. Amini, T. Weymouth, and R. Jain, "Using dynamic programming for solving variational problems in vision," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 12, no. 9, pp. 855–867, 1990.

[10] H. Bang and T. Chen, "Mining video editing rules in video streams," *Proc. of ACM Multimedia*, pp. 255–258, 2002.

[11] I. Borgand and J.Lingoes, *Multidimensional similarity structure analysis*. New York, NY: Springer-Verlag Inc., 1987.

[12] C. Bucila, J. Gehrke, D. Kifer, and W. White, "Dualminer: A dual-pruning algorithm for itemsets with constraints," 2002.

[13] J. M. Bunker, L. Wilson, M. A. Jordan, and S. C. Feinstein, "Modulation of microtubule dynamics by tau in living cells: Implications for development and neurodegeneration," *Molecular Biology of the Cell*, vol. 15, pp. 2720–2728, June 2004.

[14] Campbell and Reece, *Biology*, 6th ed. Benjamin Cummings, 2002.

[15] L. Campbell, D. Becker, A. Azarbayejani, A. Bobick, and A. Pentland, "Invariant features for 3-D gesture recognition," in *Proceedings of the Second International Conference on Automatic Face and Gesture Recognition*.

[16] V. Caselles and B. Coll, "Snakes in movement," *SIAM Journal on Numerical Analysis*, vol. 33, no. 6, pp. 2445–2456, 1996.

[17] V. Caselles, R. Kimmel, and G. Sapiro, "Geodesic active contours," *IJCV 1997*.

[18] ——, "Geodesic active contours," in *IEEE International Conference on Computer Vision*, June 1995, pp. 694–699.

[19] M. Castaud, M. Barlaud, and G. Aubert, "Tracking video objects using active contours," in *Proceedings. Workshop on Motion and Video Computing*, Dec. 2002, pp. 90–95.

[20] S. Chang, W. Chen, and H. Sundaram, "A fully automated content-based video search engine supporting spatiotemporal queries," *IEEE Trans. onCircuits and Systems for Video Technology*, vol. 8, no. 5, pp. 602–615, September 1998.

[21] Y.-L. Chang and J. Aggarwal, "Line correspondences from cooperating spatial and temporal grouping processes for a sequence of images," *Computer Vision and Image Understanding*, vol. 67, no. 2, pp. 186–201, Aug 1997.

[22] C. Chatfield, *The Analysis of Time Series: An Introduction*, sixth edition ed. CRC press, 2004.

[23] V. S. Cherkassky and F. Mulier, *Learning from Data: Concepts, Theory, and Methods*, 1st ed. John Wiley & Sons, Inc., 1998.

[24] S. Chung, J. Jun, and D. McLeod, "Mining gene expression datasets using density-based clustering," in *Proc. of CIKM 2004*, November 2004, pp. 150–151.

[25] L. Cohen, "On active contour models and balloons," *Computer Vision, Graphics, and Image Processing: Image Understanding*, vol. 53, no. 2, pp. 211–218, 1991.

[26] G. Danuser, P. Tran, and E. Salmon, “Tracking differential interference contrast diffraction line images with nanometre sensitivity,” *Journal of Microscopy, 198*, pp. 34–53.

[27] G. Danuser and C. Waterman-Storer, “Quantitative fluorescent speckle microscopy: where it came from and where it is going.” *Journal of Microscopy*, vol. 211, pp. 191–207, September 2003.

[28] H. Delingette and J. Montagnat, “Shape and topology constraints on parametric active contours,” *Computer Vision and Image Understanding: CVIU*, vol. 83, no. 2, pp. 140–171, 2001.

[29] D. Demirdjian and T. Darrell, “Constraining human body tracking,” in *Proc. of CVPR*, 2003, pp. 1071 –1078.

[30] A. Dempster, N. Laird, and D. Rubin, “Maximum likelihood from incomplete data via the em algorithm,” *Journal of the Royal Statistical Society*, vol. 34, pp. 1–38, 1977.

[31] Y. Deng and B. S. Manjunath, “Netra-V: Toward an object-based video representation,” *IEEE Trans. on Circuits and Systems for Video Technology*, vol. 8, no. 5, pp. 616–627, September 1998.

[32] A. Desai and T. J. Mitchison, “Microtubule polymerization dynamics,” *Annual Review of Cell and Developmental Biology*, vol. 13, pp. 2720–2728, November 1997.

[33] J. F. Dorn, K. Jaqaman, D. R. Rines, G. S. Jelson, P. K. Sorger, and G. Danuser, "Yeast kinetochore microtubule dynamics analyzed by high-resolution three-dimensional microscopy," *Biophysical Journal*, vol. 89, pp. 2835–2854, 2005.

[34] T. Drummond and R. Cipolla, "Real-time visual tracking of complex structures," *TPAMI 2002*.

[35] S. Eickeler and S. Muller, "Content-based video indexing of tv broadcast news using hidden markov models," *Proc. of ICASSP*, vol. 6, pp. 2997–3000, 1999.

[36] E. P. et al., "Video segmentation by map labeling of watershed segments," *TPAMI*, no. 3, pp. 326–332, 2001.

[37] F. M. et al., "Spatiotemporal segmentation based on region merging," *TPAMI*, vol. 20, no. 9, pp. 897–915, 1998.

[38] K. K. et al., "βIII-tubulin induces paclitaxel resistance in association with reduced effects on microtubule dynamic instability," *J. Biol. Chem.*, vol. 280, no. 13, pp. 12 902–12 907, Apr 2005.

[39] L. L. et al., "Extraction of 3D microtubules axes from cellular electron tomography images," in *Proc. of ICPR*, vol. 1, 2002, pp. 804–807.

[40] P. V. et al., "Tracking retrograde flow in keratocytes: News from the front." *Molecular Cell Biology*, vol. 16, pp. 1223–1231.

[41] S. H. et al., “Automatic quantification of microtubule dynamics,” in *Proc. of Int. Symp. on Biomedical Imaging: From Nano to Macro*, 2004.

[42] N. Eveno, A. Caplier, and P. Coulon, “Accurate and quasi-automatic lip tracking,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 5, pp. 706–715, May 2004.

[43] S. Feinstein and L. Wilson, “Inability of tau to properly regulate neuronal microtubule dynamics: a loss-of-function mechanism by which tau might mediate neuronal cell death.” *Biochim Biophys Acta*, vol. 36, no. 2, pp. 268–79, Jan 2005.

[44] A. Ferman and A. Tekalp, “Probabilistic analysis and extraction of video content,” *Proc. of ICIP*, vol. 2, pp. 91–95, 1999.

[45] H. Flyvbjerg, T. E. Holy, and S. Leibler, “Microtubule Dynamics: Caps, Catastrophes and Coupled Hydrolysis,” *Physical Review E*, vol. 54, no. 5, pp. 5538–5560, 1996, microtubule dynamics model.

[46] H. Flyvbjerg and E. Jobs, “Microtubule Dynamics. II. Kinetics of Self-Assembly,” *Physical Review E*, vol. 56, no. 6, pp. 7083–7099, 1997, microtubule dynamics self-assembly.

[47] A. P. Gasch and M. B. Eisen, “Research exploring the conditional coregulation of yeast gene expression through fuzzy k-means clustering,” *Genome Biology*, vol. 3, no. 11, October 2002.

[48] J. Geusebroek, A. Smeulders, and J. van de Weijer, "Fast anisotropic gauss filtering," *IEEE Transactions on Image Processing*, vol. 12, no. 8, pp. 938–943, Aug 2003.

[49] Y. Gong, L. Sin, C. Chuan, H. Zhang, and M. Sakauchi, "Automatic parsing of tv soccer programs," *Proc. of ICMCS*, pp. 167–174, 1995.

[50] R. Gonzales and R. Woods, *Digital Image Processing*, second edition ed. Prentice Hall, 2002.

[51] D. Gruber, K. Faire, and J. C. Bulinski, "Abundant expression of the microtubule-associated protein, ensconsin (E-MAP-115), alters the cellular response to taxol," *Cell Motility and the Cytoskeleton*, pp. 49:115–129, 2001.

[52] O. J. Gruss, M. Wittmann, H. Yokoyama, R. Pepperkok, T. Kufer, H. Sillj, E. Karsenti, I. W. Mattaj, and I. Vernos, "Chromosome-induced microtubule assembly mediated by tpx2 is required for spindle formation in hela cells," *Nature Cell Biology*, pp. 871–879, 2002.

[53] S. Hadjidemetriou, D. Toomre, and J. Duncan, "Tracking the motion of the outer tips of microtubules," in *Proc. of Int. Symp. on Biomedical Imaging*, 2006.

[54] S. Hadjidemetriou, D. Toomre, and J. S. Duncan, "Segmentation and 3D reconstruction of microtubules in total internal reflection fluorescence microscopy (TIRFM)," in *8th International Conference on Medical Image Computing and Computer Assisted Intervention, (MICCAI)*, 2005.

[55] C. Han, W. S. Kerwin, T. S. Hatsukami, J. Hwang, and C. Yuan, "Detecting objects in image sequences using rule-based control in an active contour model," *IEEE Transactions on Biomedical Engineering*, vol. 50, no. 6, pp. 705–710, June 2003.

[56] S. Hongeng, B. F., and N. R., "Representation and optimal recognition of human activities," in *CVPR*, vol. 1, 2000, pp. 818–825.

[57] A. Jain, A. Vailaya, and X. Wei, "Query by video clip," *Multimedia Systems*, vol. 7, pp. 369–384, 1999.

[58] E. Jaser, J. Kittler, and W. Christmas, "Hierarchical decision making scheme for sports video categorisation with temporal post-processing," in *Proc. of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, vol. 2, July 2004, pp. 908–913.

[59] M. Jiang, Q. Ji, and B. F. McEwen, "Enhancement of microtubules in em tomography," in *Proc. of Int. Symp. on Biomedical Imaging*, 2004, pp. 1123–1126.

[60] ——, "Model-based automated segmentation of kinetochore microtubule from electron tomography," in *Proc. of 26th Annual International Conference of the Engineering in Medicine and Biology Society*, 2004.

[61] ——, "Automated extraction of microtubules and their plus-ends," in *WACV/MOTION 2005*, 2005, pp. 336–341.

[62] M. Jordan and L. Wilson, “Microtbules as a target for anticancer drugs.” *Nature reviews Cancer*, vol. 4, pp. 253–265, Apr 2004.

[63] Z. Kalafatic, S. Ribaric, and V. Stanisavljevic, “A system for tracking laboratory animals based on optical flow and active contours,” in *International Conference on Image Analysis and Processing*, Sept. 2001, pp. 334–339.

[64] A. Kale, A. Sundaresan, A. N. Rajagopalan, N. P. Cuntoor, A. K. Roy-Chowdhury, V. Kruger, and R. Chellappa, “Identification of humans using gait,” *IEEE Transactions on Image Processing*, vol. 13, pp. 1163–1173, Sep. 2004.

[65] K. Kamath and M. A. Jordan, “Suppression of microtubule dynamics by epothilone B is associated with mitotic arrest.” in *Cancer Res.*, vol. 63, 2003, pp. 6026–6031.

[66] B. Kamgar-Parsi, “Algorithms for matching 3D line sets,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 5, pp. 582–593, May 2004.

[67] M. Kass, A. Witkin, and D. Terzopoulos, “Snakes - active contour models,” *International Journal of Computer Vision*, vol. 1, no. 4, pp. 321–331, 1988.

[68] S. Khan and M. Shah, “Object based segmentation of video using color, motion and spatial information,” in *Proc. of CVPR*, vol. 2, 2001, pp. 746–751.

[69] R. Kimmel and A. Bruckstein, “Regularized laplacian zero crossings as optimal edge integrators,” *International Journal of Computer Vision*, vol. 53, no. 3, pp. 225–243, 2003.

[70] R. Kimmel, *Numerical Geometry of images: Theory, algorithms and applications*. Springer, 2004.

[71] V. Kobla, D. Doermann, and C. Faloutsos, “Videotrails : Representing and visualizing structure in video sequences,” in *ACM Multimedia*, 1997, pp. 335–346.

[72] M. Kurokawa, T. Echigo, A. Tomita, J. Maeda, H. Miyamori, and S. Iisaku, “Representation and retrieval of video scene by using object actions and their spatio-temporal relationships,” *Proc. of ICIP*, vol. 2, pp. 86–90, 1999.

[73] R. Kutka and S. Stier, “Extraction of line properties based on direction fields,” *IEEE Transactions on Medical Imaging*, vol. 15, no. 1, pp. 51–58, Feb 1996.

[74] V. Levi, A. Serpinskaya, E. Gratton, and V. Gelfand, “Organelle transport along microtubules in xenopus melanophores: evidence for cooperation between multiple motors,” *Biophysical Journal*, vol. 90, no. 1, pp. 318–327, 2006.

[75] T. Lindeberg, “Edge detection and ridge detection with automatic scale selection,” in *Computer Vision and Pattern Recognition*, June 1996, pp. 465–470.

[76] J. Liu, W. Hwang, M. Chen, J. Tsai, and C. Lin;, "Wavelet based active contour model for object tracking," in *Proceedings. International Conference on Image Processing*, vol. 3, Oct. 2001, pp. 206–209.

[77] T. Liu and J. Kender, "A hidden markov model approach to the structure of documentaries," *Proc. of workshop CBAIVL in CVPR*, pp. 111–115, 2000.

[78] W. Ma and B. Manjunath, "A texture thesaurus for browsing large aerial photographs," *Journal of the American Society of Information Science*, vol. 49, no. 7, pp. 633–648, 1998.

[79] P. Maddox, A. Straight, P. Coughlin, T. Mitchison, and E. Salmon, "Direct observation of microtubule dynamics at kinetochores in xenopus extract spindles: implications for spindle mechanics," *Journal Cell Biology*, vol. 162, no. 3, pp. 377–382, 2003.

[80] V. Malikov, E. N. Cytrynbaum, A. Kashina, A. Mogilner, and V. Rodionov, "Centering of a radial microtubule array by translocation along microtubules spontaneously nucleated in the cytoplasm," *Nature Cell Biology*, 2005.

[81] R. Marti, R. Zwiggelaar, and C. Rubin, "Tracking mammographic structures over time," in *British machine vision conference*, 2001.

[82] J. Menendez and S. Velastin, "A method for obtaining neural network training sets in video sequences," *Int. Workshop on Visual Surveillance*, pp. 69–75, 2000.

[83] H. Miyamori and S. Iisaku, "Video annotation for content-based retrieval using human behavior analysis and domain knowledge," *Proc. of AFGR*, pp. 320–325, 2000.

[84] M. I. Molodtsov, E. A. Ermakova, E. E. Shnol, E. L. Grishchuk, J. R. McIntosh, and F. I. Ataullakhanov, "A molecular-mechanical model of the microtubule," *Biophysical journal*, vol. 88, pp. 3167–3179, 2005, microtubule dynamics model.

[85] D. Mukherjee, N. Ray, and S. Acton, "Level set analysis for leukocyte detection and tracking," *IEEE Transactions on Medical Imaging*, vol. 13, no. 4, pp. 562–572, April 2004.

[86] M. Naphade and T. Huang, "A probabilistic framework for semantic video indexing, filtering, and retrieval," *IEEE Trans. on Multimedia*, vol. 3, no. 1, pp. 141–145, March 2001.

[87] C. Ngo, T. Pong, and H. Zhang, "On clustering and retrieval of video shots," in *ACM Multimedia*, 2001, pp. 51–60.

[88] S. Osher and J. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on hamilton–jacobi formulations," *Journal of Computational Physics*, vol. 79, pp. 12–49, 1988.

[89] S. Osher and R. Fedkiw, *Level set methods and dynamic implicit surfaces*. Springer, applied mathematical sciences, vol. 153.

[90] N. Paragios and R. Deriche, "Geodesic active contours and level sets for the detection and tracking of moving objects," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 22, no. 3, pp. 266–280, March 2000.

[91] S. Ponce-Dawson, J. E. Pearson, and W. Reynolds, "Geometric features of microtubule dynamics."

[92] L. Rabiner and B. H. Juang, *Fundamentals of Speech Recognition*. Prentice-Hall, 1993.

[93] D. Ramanan and D. Forsyth, "Finding and tracking people from the bottom up," in *Proc. of CVPR*, vol. 2, 2003, pp. 467–474.

[94] C. K. Reiter-Funk and D. P. Dohrman, "Chronic ethanol exposure increases microtubule content in pc12 cells," *BMC Neuroscience*, p. 6:16, 2005.

[95] C. J. V. Rijsbergen, *Information retrieval*, 2nd ed. London: Butterworths, 1979.

[96] M. Rochery and I. J. J. Zerubia, "Higher order active contours and their application to the detection of line networks in satellite imagery," in *Proceedings of 2nd IEEE Workshop on Variational, Geometric and Level Set Methods in Computer Vision at ICCV*, Oct 2003.

[97] V. Rodionov, E. Nadezhdina, and G. Borisy, "Centrosomal control of microtubule dynamics," in *Proc. Natl. Acad. Sci.*, vol. 96, January 1999, pp. 115–120.

[98] D. Saur, Y. Tan, S. Kulkarni, and P. Ramadge, "Automated analysis and annotation of basketball video," *Proc. of SPIE*, vol. 3022, pp. 176–187, 1997.

[99] J. Sethian, *Level Set Methods and Fast Marching Methods*. Cambridge Univ. Press, 1999.

[100] K. Shafique and M. Shah, "A noniterative greedy algorithm for multiframe point correspondence," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27, no. 1, pp. 51–65, Jan 2005.

[101] C. Steger, "An unbiased detector of curvilinear structures," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 2, pp. 113–125, Feb 1998.

[102] G. Sudhir, J. Lee, and A. Jain, "Automatic classification of tennis video for high-level content-based retrieval," *Proc. of CBAIVL in CVPR*, pp. 81–90, 1998.

[103] X. Sun and B. S. Manjunath, "Panoramic capturing and recognition of human activity," vol. 2, pp. 813–816, Sep. 2002.

[104] J. Tang and S. T. Acton, "Vessel boundary tracking for intravital microscopy via multiscale gradient vector flow snakes," *IEEE Transactions on Biomedical Engineering*, vol. 51, no. 2, pp. 316–324, Feb 2004.

[105] H. Tao, H. Sawhney, and R. Kumar, "Object tracking with bayesian estimation of dynamic layer representations," *IEEE TPAMI*, vol. 24, no. 1, pp. 75–89, January 2002.

[106] J. Tesic, S. Newsam, and B. Manjunath, "Scalable spatial event representation," *Proc. of ICME*, pp. 229–232, 2002.

[107] R. Tournebize, A. Popov, K. Kinoshita, A. J. Ashford, S. Rybina, A. Pozniakovsky, T. U. Mayer, C. E. Walczak, E. Karsenti, and A. A. Hyman, "Control of microtubule dynamics by the antagonistic activities of xmap215 and xkcm1 in xenopus egg extracts," in *Nature cell biology*, vol. 2, January 2000, pp. 13–19.

[108] F. Tupin, H. Maitre, J. Mangin, J. Nicolas, and E. Pechersky, "Detection of linear features in sar images: application to road network extraction," *IEEE Transactions on Geoscience and Remote Sensing*, vol. 36, no. 2, pp. 434–453, Mar 1998.

[109] N. Vaswani, A. Roy-Chowdhury, and R. Chellappa, ""Shape activity": a continuous-state HMM for moving/deforming shapes with application to abnormal activity detection," in *Transactions on Image Processing, IEEE*, vol. 14, 2005, pp. 1603–1616.

[110] C. Vogler and D. Metaxas, "ASL: Recognition based on a coupling between hmms and 3D motion analysis," in *Proceedings of the International Conference on Computer Vision*.

[111] L. Xie, S.-F. Chang, A. Divakaran, and S. Huifang, "Feature selection for unsupervised discovery of statistical temporal structures in video." in *Proc. of ICIP*, vol. 1, 2003, pp. 29–32.

[112] C. Xu and J. L. Prince, "Gradient vector flow: a new external force for snakes," in *Computer Vision and Pattern Recognition*, June 1997, pp. 66–71.

[113] A.-M. C. Yvon, P. Wadsworth, and M. A. Jordan, "Taxol suppresses dynamics of individual microtubules in living human tumor cells," *Molecular biologhy of the cell*, vol. 10, no. 4, pp. 947–959, April 1999.

[114] O. Zaiane, J. Han, Z. Li, J. Chiang, and S. Chee, "Multimedia-miner: A system prototype for multimedia data mining," *Proc. of ACM-SIGMOD*, 1998.

[115] I. Zaliapin, I. Semenova, A. Kashina, and V. Rodionovy, "Multiscale trend analysis of microtubule transport in melanophores," *Biophysical Journal*, vol. 88, pp. 4008–4016, June 2005.

[116] D. Zhong and S. Chang, "Structure analysis of sports video using domain models," *Proc. of ICME*, pp. 920–923, 2001.

[117] W. Zhou, A. Vellaikal, and C. Kuo, "Rule-based video classification system for basketball video indexing," in *Proceedings of the 2000 ACM workshops on Multimedia*, 2000, pp. 213–216.

[118] A. Zhu, J. Fan, M. Hacid, and A. Elmagarmid, "Classminer: Mining medical video for scalable skimming and summerization," *Proc. of ACM Multimedia*, pp. 79–80, 2002.

[119] X. Zhu, W. Aref, J. Fan, A. Catlin, and A. Elmagarmid, "Medical video mining for efficient database indexing, management," *Proc. of ICDE*, 2003.

[120] C. Zimmer, E. Labruyere, V. Meas-Yedid, N. Guillen, and J. Olivo-Marin, "Segmentation and tracking of migrating cells in videomicroscopy with parametric active contours: a tool for cell-based drug testing," *IEEE Transactions on Medical Imaging*, vol. 21, no. 10, pp. 1212–1221, Oct. 2002.