



Theses and Dissertations

2010-03-12

Automatic Identification and Tracking of Retraction Fibers in Time-Lapse Microscopy

Meher Talat Shaikh
Brigham Young University - Provo

Follow this and additional works at: <https://scholarsarchive.byu.edu/etd>



Part of the [Electrical and Computer Engineering Commons](#)

BYU ScholarsArchive Citation

Shaikh, Meher Talat, "Automatic Identification and Tracking of Retraction Fibers in Time-Lapse Microscopy" (2010). *Theses and Dissertations*. 2093.
<https://scholarsarchive.byu.edu/etd/2093>

This Thesis is brought to you for free and open access by BYU ScholarsArchive. It has been accepted for inclusion in Theses and Dissertations by an authorized administrator of BYU ScholarsArchive. For more information, please contact scholarsarchive@byu.edu, ellen_amatangelo@byu.edu.

Automatic Identification and Tracking of Retraction
Fibers in Time-Lapse Microscopy

Meher Talat Shaikh

A thesis submitted to the faculty of
Brigham Young University
in partial fulfillment of the requirements for the degree of
Master of Science

James K. Archibald, Chair
Chia-Chi Teng
Dah-Jye Lee

Department of Electrical and Computer Engineering
Brigham Young University
April 2010

Copyright © 2010 Meher Talat Shaikh

All Rights Reserved

ABSTRACT

Automatic Identification and Tracking of Retraction Fibers in Time-Lapse Microscopy

Meher Talat Shaikh

Department of Electrical and Computer Engineering

Master of Science

Digital image processing is widely used in the field of time-lapse microscopy and biological research to provide statistical data of cellular dynamics. The data can provide more comprehensive understanding of the molecular phenomenon. Further, digital image processing enables rapid and consistent quantification of qualitative observations. The image processing model examined here provides a study to identify structures called retraction fibers (RFs) that are formed during epithelial-mesenchymal transition (EMT) [1], an important developmental process which also occurs during cancer metastasis. Quantifying RF formation is an important task for biologists studying cellular regulation of EMT. This thesis work uses digital image processing and computer vision algorithms to detect and track each RF in image sequences of cells undergoing EMT that are captured using time-lapse microscopy. The algorithms isolate the RFs with reasonable precision. Statistical information is generated about these automatically detected RFs, such as the number formed during a particular time window, lifetime of each, and their geometric dimension. This information can in turn be used by biologists to quantitatively measure the extent of EMT under different test conditions. Biologists feel that the information thus obtained may help clarify the molecular interactions of cell migration and will aid in developing methods of preventing cancer metastasis. Experimental results show that this methodology has significant potential in helping biologists determine RF behavior during EMT.

Keywords: Epithelial-mesenchymal transition, digital image processing, cancer metastasis, retraction fibers

ACKNOWLEDGMENTS

I greatly appreciate the guidance, time and effort put in by my advisors Dr. James Archibald and Dr. Chia-Chi Teng to bring this work to a reasonable analysis. I am very grateful for their encouragement and thoughtful advice during the execution of this project. I extend my sincere thanks to Dr. Dah-Jye Lee for taking a keen interest in patiently evaluating my project, and providing constructive feedback. I am thankful to Dr. Clark Taylor for introducing me to the field of digital image processing and the image processing libraries. I owe sincere thanks to Rebecca who answered all my questions regarding the thesis data. I am thankful to all my teachers at Brigham Young University who inspired me to execute this thesis as best as I can. I am greatly indebted to my husband Mujahed and children Arub, Lamis and the next on the way who patiently waited for the thesis work to get completed so that I could have enough time for them. I could not have been able to maintain a balance between family and the project work without their help and understanding. I am also thankful to my parents whose prayers have helped me as I was learning about the challenging aspects of this project. I would like to acknowledge my friends Sariah Hillam and Alicia Rawlins who were a big help with my school aged children. I also want to acknowledge Lloyd and Dixie Baker for continually inspiring me to remain focused in my work. I would like to sincerely acknowledge the contributions of all who were directly or indirectly involved with this project. This project would not have been completed without their support.

TABLE OF CONTENTS

LIST OF TABLES	vii
LIST OF FIGURES	ix
1 INTRODUCTION.....	1
1.1 Epithelial-mesenchymal transition (EMT)	1
1.2 Computer vision.....	3
1.3 Significance of computer vision in RF identification and tracking	4
1.4 Challenges of tracking	5
1.5 Problem.....	6
1.6 Outline	7
2 BACKGROUND OF MOLECULAR TRACKING	9
2.1 Tracking in molecular imaging.....	9
2.2 Cell tracking.....	10
3 IMAGE PRE-PROCESSING	15
3.1 Smoothing.....	15
3.2 Gaussian smoothing.....	16
3.3 Thresholding	16
3.4 Discarding cell structure regions from the binary image.....	19
4 DETECTION AND TRACKING OF RFS.....	23
4.1 Connected components.....	23
4.2 RFS extraction.....	24
4.3 Black Hat transform.....	26
4.4 Hough transform.....	28
4.4.1 Hough line transform	29

4.4.2	Post-processing of the detected lines	31
4.5	Tracking RFs.....	32
4.5.1	Tracking methodology	34
4.6	Detection and tracking example	35
5	RESULTS	37
5.1	Edge image for line detection	39
5.2	Effect of post processing.....	42
5.3	λ selection	43
5.4	System performance across different experiments	46
5.5	Tracking results.....	47
5.6	RFs lifetime information.....	51
6	CONCLUSION	53
6.1	Identification of RFs	53
6.2	Tracking of RFs	53
6.3	Future work.....	54
	REFERENCES.....	57

LIST OF TABLES

Table 5-1 RFs detection and tracking using the 3 approaches.....	40
Table 5-2 Effect of post processing strategy for RFs	43
Table 5-3 System performance for the three image sequences.....	47
Table 5-4 Number of RFs tracked with a unique/multiple ID	49
Table 5-5 Lifetime information statistics of RFs.....	51

LIST OF FIGURES

Figure 1-1 Retraction fibers formed during EMT.....	2
Figure 1-2 Typical EMT sequence (3 consecutive frames).....	3
Figure 3-1 Global thresholding.....	17
Figure 3-2 Left: Original image, Right: Image after applying adaptive threshold.....	19
Figure 3-3 Processing example for maintaining thinner regions.....	20
Figure 3-4 Binary image of Figure 3-2 with cell pixels removed.....	21
Figure 4-1 Contours of size 13.....	25
Figure 4-2 RFs extraction in an image.....	26
Figure 4-3 RF extraction.....	30
Figure 4-4 Strategy to reduce FPs.....	32
Figure 4-5 Eight RFs detected in bottom-hat image.....	33
Figure 4-6 Six RFs detected as a result of post-processing step.....	33
Figure 4-7 Example of two RFs tracked in a three-frame sequence.....	34
Figure 4-8 RFs detection and tracking.....	36
Figure 5-1 Precision of RF detection in individual frames of an image sequence.....	42
Figure 5-2 RFs detected with line threshold 20.....	44
Figure 5-3 RFs detected with line threshold 15.....	44
Figure 5-4 RFs detected with line threshold 20 in Laplacian transformed image.....	45
Figure 5-5 RFs detected with line threshold 15 in Laplacian threshold image.....	45
Figure 5-6 Effect of line threshold on precision.....	46
Figure 5-7 Example of adaptive thresholding contributing to missing RFs.....	48
Figure 5-8 Examples of overlapping rectangles providing unique IDs for RFs.....	49
Figure 5-9 Examples of overlapping rectangles providing multiple IDs for RFs.....	50

Figure 5-10 Example tracking of reduced lifetime of RFs52
Figure 6-1 RFs lost at adaptive thresholding stage.....55

1 INTRODUCTION

This work describes a computer-based approach to identify and track thin molecular structures called retraction fibers (RFs) in a time lapse sequence of microscopic images. The RFs are an important phenomenon that occurs during cell metastasis known as epithelial-mesenchymal transition (EMT). In this chapter we introduce EMT and describe the significance of our work related to the RFs that appear during EMT.

1.1 Epithelial-mesenchymal transition (EMT)

Epithelial-mesenchymal transition (EMT) is an important process in cell biology that takes place during embryonic development as well as in cancer metastasis. A typical epithelium is a sheet of cells where the cell-cell junctions and adhesions between neighboring cells hold them together and inhibit the movement of individual cells. During EMT, cells disrupt cell-cell junctions because of loss of cell adhesion and leave the epithelium as individual migratory cells. As cells separate, retraction fibers (RFs) form as cell membranes are stretched between separating cells. Figure 1-1 shows a typical retraction fiber formed during EMT. The cells eventually separate and migrate to other locations, thereby assuming the state of migration, and thus spreading over larger areas. This EMT process can be regulated by zyxin, a known adherens junction protein [2]. Since proteins of this type help in strengthening the attachments between adjacent cells, zyxin also changes the rate of EMT [3]. Understanding how EMT is regulated at the molecular level is therefore an important question in cell biology. Biologists would like to

explore the role of zyxin during separation of cell-cell contacts during EMT. Since RFs appear before the cells break and migrate, biologists would like to study their behavior. This study may help devise prevention and control of life-threatening diseases such as cancer.

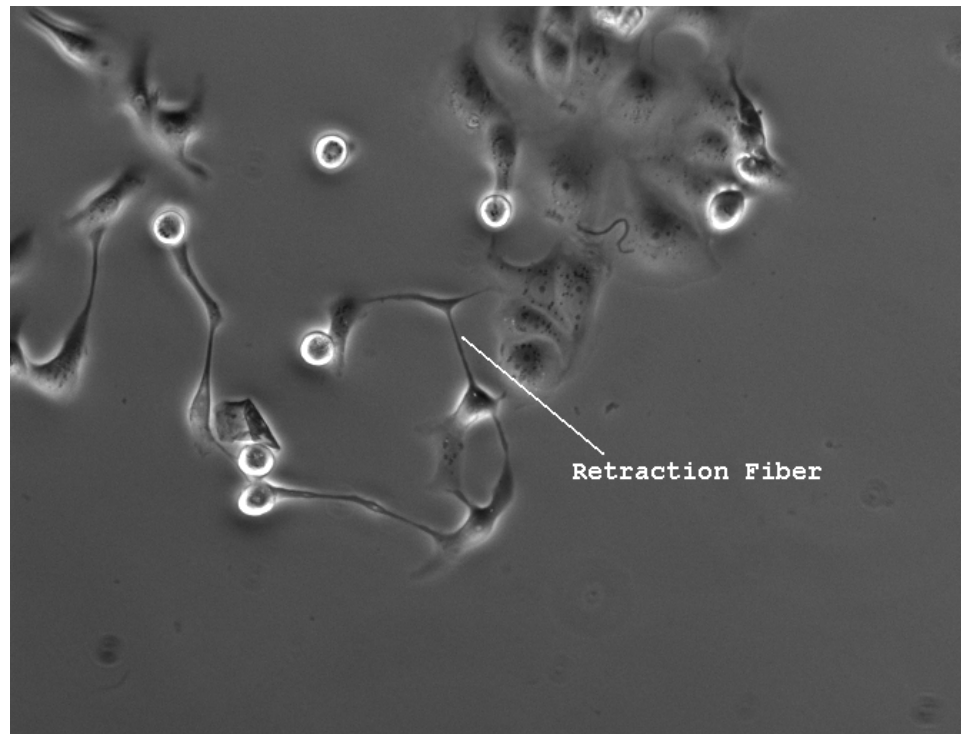


Figure 1-1 Retraction fibers formed during EMT

The dynamic EMT process is captured by biologists using time-lapse microscopy at a 2-minute interval. Figure 1-2 shows a typical EMT sequence of 3 frames. Currently, to study the effect of zyxin on EMT, biologists capture the retraction fiber information manually. The researchers then examine the RFs in each of the time-lapse microscopic images. As seen in Figure 1-1, retraction fibers appear as thin molecular structures stretched between cells.

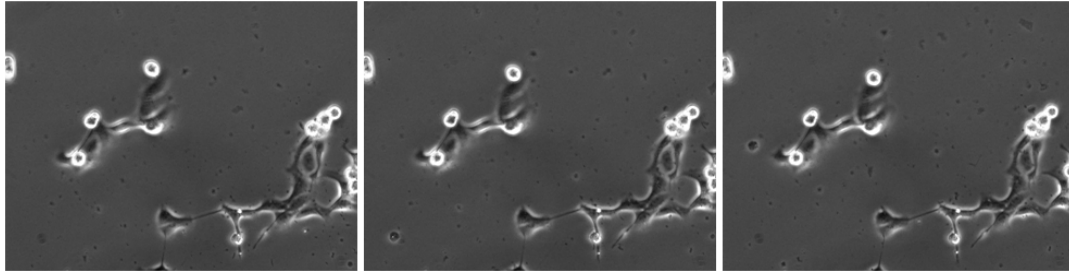


Figure 1-2 Typical EMT sequence (3 consecutive frames)

1.2 Computer vision

Computer vision involves the process of obtaining information from images, often in a manner that mimics the human vision system. It is used extensively for important applications in many fields. With evolving molecular imaging techniques, computer vision has become increasingly attractive to researchers in the field of biomedical analysis. The massive amounts of imaging data collected for study call for a computerized methodology of analysis.

Given an image or sequence of images, computer vision involves isolating critical parts of the image under study from other image parts. This is generally referred to as feature extraction in image processing terminology. In this work, computer vision is used to detect the RFs and to track their lifetimes through a time-lapse microscopic image sequence. Each image sequence consists of 300-400 frames captured during EMT. In this time interval, the RFs between the cells appear at some time-point and then they may disappear at some other time-point, exhibiting seemingly random motion. This work focuses on the use of computer vision to detect these dynamic RFs and to determine statistics of their lifetimes. The computer vision algorithms implemented in this work to detect RFs include image blurring, thresholding, locating thin molecular structures that are potential RF fibers, and the Hough transform. The RF

dynamics are determined using the space and time information (generally referred to as spatiotemporal statistics) of the RF phenomenon.

1.3 Significance of computer vision in RF identification and tracking

Currently, to determine the RF dynamics to study the effect of zyxin on EMT, biologists capture the retraction fiber information manually. For example, researchers at BYU examine each image in the sequence of the time-lapse microscopic images, and use the Olympus SlideBook [4] software package to manually measure the length of each retraction fiber at each time-point. After examining the entire EMT image sequence, researchers have statistics such as the number of unique RFs that appeared over the EMT sequence and the maximum length attained by each RF before it breaks. This manual process is time-consuming and tedious, imposing a practical limit on the amount of data that can be collected. The quantification and analysis of thousands or more RFs in different EMT sequences would require significant human time and effort to determine the RFs behavior.

There is also variability and inconsistency between researchers in identifying RFs. The objective of this thesis is to explore the feasibility of creating a tool that can identify and track the lifetime of these RFs automatically, using computer vision algorithms. With such a tool, retraction fiber data for parental cells and cells expressing zyxin mutants during EMT could be rapidly and reliably produced by the tool and used by biologists for statistical analysis to determine differences in the RF behavior. The tool would thus provide a means to quantify the effect of altering zyxin function during EMT, thus leading to greater understanding of the role of zyxin in regulation of EMT and cancer metastasis. Ideally, computer vision techniques would enable a fast and consistent way of identifying the RFs in these image frames in such a way that

the results could easily be reproduced. The tool would help to decrease reliance on human expertise and provide a more cost effective solution for the study of EMT. An improved understanding of this phenomenon could potentially result in the prevention and control of cancer and other life threatening metastasis.

1.4 Challenges of tracking

Although automatic detection and tracking is the solution for a vast amount of quantification and analysis, achieving high precision is challenging. Human vision is able to perceive and process a significant amount of information from images within seconds. Additionally, the human brain has the ability to draw many inferences and apply background knowledge to the environment under study. Using training-data, a computer can be made to draw inferences and use background knowledge, but the amount of data that can be fed as background knowledge still has a limit. Images that appear obvious and straightforward to human vision are often very difficult for a computer to process. While human vision can easily delineate the desired features from an image, it is an arduous task for the computer to manipulate the image pixels and to obtain an approximately equivalent perception. For example, in the image sequences used in this thesis, histograms of grayscale values show unimodal distribution, and this makes it difficult to isolate the desired features of the image based on intensity of pixels. Further, the task to identify the true RFs from the sea of molecular structures that are detected as potential RFs is not trivial. In the end, some level of inaccuracy needs to be tolerated to achieve the speed, reproducibility and cost effectiveness of a practical computer vision system based on these techniques.

Meijering et al. [7] note the fundamental problems and limitations in achieving robustness and accuracy in tracking and motion analysis in images obtained by light microscopy. Accordingly, the performance of computer vision is greatly hampered by the following three factors: the limited spatial resolution of the microscope because of diffraction, the signal-to-noise ratio, and the large variability of biological image data. They also note that the quality of images may differ even in the same experiment. In the work described in this thesis, each of the above factors was experienced to a certain degree. The input microscopic images are of variable illumination. Further, even the best implementation could not track all the RFs that were detected manually, leading to a loss of critical data. Some RFs have almost negligible contrast with the background and are lost in image thresholding. Further, in automatic tracking there are examples where separate and distinct RFs are identified incorrectly as the same fiber. Thus, it is desirable but difficult to minimize these undesired tracking side effects.

1.5 Problem

The goal of the current work is to explore the feasibility of developing an image processing program that does the following:

- Identify each retraction fiber as it is formed during EMT,
- track each fiber through its lifetime (i.e. for each fiber, identify the corresponding fiber, if any, across all frames), and
- provide statistics for the life span of each fiber.

1.6 Outline

The following chapters discuss the details of the implementation of automatic detection and tracking of RFs. We begin with background work in automatic detection and tracking of molecular structures in cell biology (Chapter 2). Here we discuss the challenges involved in molecular tracking and bring to light some important identification and tracking approaches. The literature provides a general understanding of the computer vision task that we try to accomplish.

Chapter 3 covers image preprocessing algorithms that are used to extract the relevant information from the microscopic images while suppressing the irrelevant information. In Chapter 4 we describe in depth our entire approach towards the solution of the problem undertaken.

Chapter 5 reports the results of our implementation. And finally we conclude our findings in Chapter 6 with future work that could be taken up to enhance our system performance.

2 BACKGROUND OF MOLECULAR TRACKING

The study of molecular dynamics using a microscopic time-lapse imaging technique is common. The imaging technique produces a huge amount of data that calls for automatic quantification of the molecular motion analysis. Automatic quantification involves first locating the molecular structures in the images and then linking them across the images using spatial and temporal information. This process is generally referred to as tracking. In the past, a significant amount of work has been done using computer vision algorithms in the area of tracking cells and various other molecular moving structures that are captured using time-lapse microscopy [5] [6] [7] [8] [9] [10]. These automatic methods are hampered by typical challenges associated with the analysis of molecular dynamics. These include low contrast, noise, variable illumination, and random motion in the microscopic images. These obstacles are also present in our input images. In this chapter we briefly summarize molecular tracking literature that provides useful insight towards a solution for this current work.

2.1 Tracking in molecular imaging

In their survey, Meijering et al. [7] cover the current state of affairs in the field of tracking in biological molecular imaging. They describe the basic principles of visualizing molecular dynamics in living cells and detail how light microscopy [11] combined with time-lapse imaging provides powerful tools to study these dynamics. In their work they also provide some examples of molecular dynamics studies to emphasize how the studies generate vast

amounts of image data, where the dynamics of hundreds or thousands of molecular structures need to be analyzed quantitatively and as completely as possible to reveal desired information. They describe how the limited spatial resolution of the microscope, noise, and large variability of biological image data impose limitations on achieving robustness and high accuracy in tracking. They discuss the significance of computer vision in analyzing the biological image data and detail a general tracking technique for molecular dynamics. Finally, they point out that any successful tracking technique would involve image preprocessing, detecting the desired features, linking them together, and analyzing them. Not surprisingly, these general steps also form the backbone of our proposed approach. Various alternatives were explored in this thesis work for each of these phases. Our approach for each phase is based on the shape, structure, and behavior of RFs, the identification of which forms the central focus of this study.

In most computer vision tracking problems, the effectiveness of an approach largely depends on the segmentation process used for molecular structure detection. Segmentation is the process of partitioning a digital image into background and foreground objects, where the foreground objects are the parts of the image that are under study. The goal of segmentation is to simplify a given image into something that is more straightforward to analyze.

2.2 Cell tracking

In the field of molecular tracking, the tracking of cells and cell population in time-lapse microscopic images has gained tremendous attention in recent years. In the following sections we will review some of the automatic detection and tracking systems that have been used for cell identification and tracking. While the molecular structures identified in these systems are cells

rather than RFs, the tracking systems provide insight into approaches that are feasible for molecular tracking.

Li et al. [5] present a computer vision based system to automatically detect and track the behavior of individual stem cells (cells that can reproduce themselves) in expanding populations. The system provides the statistics of cell movement, division, quiescence and death of every cell in a population in real time. The detection is based on the observation that the cells that undergo mitosis have an initial increase in brightness and circularity, accompanied by a decrease in size. This approach uses machine-learning to detect spatiotemporal mitosis events. As training data, the system identifies both positive and negative examples of mitosis events from an image sequence.

Shi et al. [6] propose using a fast level set implementation for real time tracking of moving objects. This is a region-based tracking model that uses tracking results from the last frame as the initial curves and then evolves each curve according to an equation that minimizes energy to locate the objects in the current frame. The method is well suited to track region boundaries that take random shapes. Li et al. [8] use the approach proposed in [6] to successfully track cell population. They employ the following modules to achieve this purpose:

- Cell detector: detects and labels candidate cell regions in input image
- Cell tracker: propagates cell regions and identities across frames
- Dynamic filter: performs prediction and filtering of cell motion dynamics using a Kalman filter
- Track arbitrator: manages the tracking task of new cells and dead/departed cells and establishes cell lineages

In our project, since the RFs are rather straight, we employed the well-understood Hough transform to detect the lines and to track RFs based on where in image space these lines are. This part of our implementation is analogous to the cell detection and cell tracking of Li et al. [8].

House et al. [9] propose another approach for tracking cell populations. It uses Bayesian algorithms to detect and track large numbers of cells in the presence of clutter and to identify cell division. In its first step it uses an adaptive threshold to produce a binary image in which cells and clutter are segmented from the background. In this binary image, groups of pixels under a certain threshold are classified as clutter and disregarded in further processing. We have implemented a similar technique (discussed later) that uses an adaptive threshold to segment the cells and RFs from the background.

Most methods in molecular tracking start by completely segmenting the desired molecular structures in each frame and then attempting to track these molecular structures from one image to the next. This approach helps to reduce the complexity of tracking. Intermediate results can be checked and used to improve each step separately. Some recent approaches establish the task of detection and tracking in a single step to improve segmentation performances [12] [13]. Primet [10] combines the advantages of both approaches by using a two-step process so that intermediate results can be checked, while retaining the idea of simultaneous segmentation and detection. In the first step it uses “over-segmentation”, a process to partition the image domain into background domain and small regions called blobs, such that any cell of the image is a union of connected blobs where any blob belongs to exactly one cell.

The major differences between the above implementations and our approach are that the retraction fiber we hope to identify is not circular and does not divide, although it appears at some time-point, extends during EMT and disappears at another time-point. Hence our statistics

are different than the approaches mentioned above. In our implementation we do not have trained classifiers for machine learning because the RFs we process may attain random lengths in random directions in variable intensities during the EMT process. This is in contrast to the cell mitosis event that may divide at any orientation but does not attain random length. In our case, we found it was not feasible to collect positive and negative samples of RFs to discriminate them from the remaining structures. Further, the RFs do not reproduce. In a given colony of cells, the number of RFs that may appear are not as numerous as cells that are reproduced whose lineage needs to be tracked, hence we employ simpler methods to capture RF statistics.

3 IMAGE PRE-PROCESSING

Images that are processed for certain feature analysis often contain more details than the features that are being analyzed. Further, the images may contain noise, artifacts, and other undesired features. Image pre-processing is an operation generally carried out in many computer vision applications to enhance important image features while suppressing image information and features that are irrelevant to the analysis task. In other words, the result of image pre-processing is an image with information content that will be further processed while disregarding the unwanted features. In our implementation, the features that we focus on and analyze are RFs, and the time-lapse microscopic images that we process are of cell colonies (group of cells) that undergo EMT. The images contain cells and, depending on the time-point, may also contain RFs. The goal of pre-processing in our approach is to suppress noise, disregard the cells, and obtain images that could show potential RFs. It involves Gaussian blurring, thresholding, extracting image regions that represent connecting areas between cells, and disregarding the cell regions. The following sections illustrate each of these steps in detail.

3.1 Smoothing

A smoothing operation is applied to an image to reduce noise. Smoothing is often referred to as blurring. Noise in an image is generally characterized as any high frequency component. Hence, the effect of smoothing is to filter out the high frequency components in an image. The filters for noise removal are classified as linear filters and non-linear filters [14].

Both are widely used based on the nature of the application. In each category there are many techniques that are frequently used. Gaussian smoothing is one example of a linear filter, while the median filter is an example of a non-linear filter. In our project we are dealing with grayscale images, and we would like to retain the thin lines that correspond to RFs during noise removal. We experimented with different filters before deciding to use a Gaussian filter which provided the best results for our task. For example, the median filter damages thin lines in the image, so it was not used.

3.2 Gaussian smoothing

The Gaussian filter performs a weighted average of surrounding pixels based on the Gaussian distribution function. For a 2D image the Gaussian function is given by

$$G(x, y) = \frac{e^{-(x^2+y^2)/2\sigma^2}}{2\pi\sigma^2}, \quad (3.1)$$

where x is the distance from the image origin in the horizontal axis, y is the distance from the image origin in the vertical axis, and σ is the standard deviation of the Gaussian distribution. The Gaussian filtering is done by convolving each point in the input image with a Gaussian function to produce the output image. To smooth the image frames of our time-lapse microscopic images we use the `cvSmooth` function in OpenCV with smooth type `CV_GAUSSIAN` [15]. A Gaussian kernel of filter window size 3x3 (neighborhood pixel window size) is used in our current approach.

3.3 Thresholding

Thresholding is an image segmentation technique by which we categorically accept or reject the pixels in an image based on the information that we desire to retain. It transforms the

input image to a simpler form for analysis. Thresholding is a process of transforming the input image f to an output image g such that

$$g(x, y) = \begin{cases} 1 & f(x, y) \geq T \\ 0 & f(x, y) < T \end{cases}, \quad (3.2)$$

where T is the threshold for pixel grayscale value. In image g , the pixel at location (x, y) is assigned either value 0 or a 1 depending on the grayscale value at location (x, y) in the image f . However, this assumes a single, fixed threshold is used for the entire image. The microscopic images, as described earlier, suffer from variable illumination and low contrast; hence, a single global threshold for the entire image is not the best approach for extracting features present in the images. Figure 3-1 shows the result of using a global threshold for a sample image.

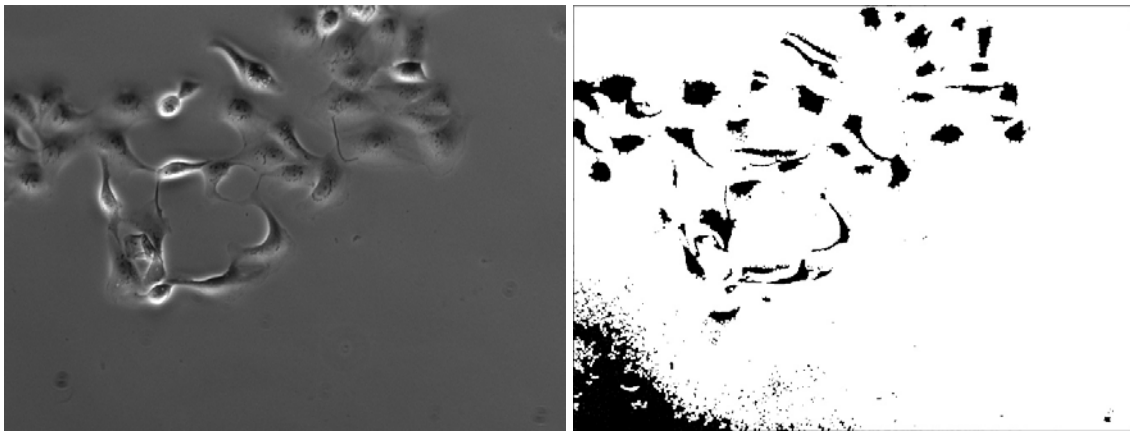


Figure 3-1 Global thresholding

The undesired effects of global thresholding are obvious in the bottom left corner of the image on the right of Figure 3-1. (The approach has identified features where no real features exist.) Therefore, we use thresholding techniques where the threshold T is position dependent and is given as:

$$T = T(f, f_c), \quad (3.3)$$

where f is the image, and f_c is a distinct portion of f over which a given threshold value is used.

Segmentation using such variable thresholds across the image is called *variable thresholding* or *adaptive thresholding*. For thresholding the microscopic images, we used a simple and fast adaptive thresholding algorithm proposed by Bradley and Roth [16] that takes into account spatial variations in illumination. The algorithm uses *integral images* to compute a different threshold value for each pixel in the image; an integral image is a tool to compute the sum of a particular image function over a rectangular region of the image. To compute the integral image, the approach suggested by Bradley and Roth [16] stores at each integral image location, $I(x,y)$, the sum of all $f(x,y)$ terms to the left and above the pixel (x,y) . This is accomplished in linear time using the following equation for each pixel:

$$I(x, y) = f(x, y) + I(x - 1, y) + I(x, y - 1) - I(x - 1, y - 1). \quad (3.4)$$

In the next step, for each pixel location of an image array, the approach then uses the integral image and computes the sum function for a specified rectangle with upper left corner (x_1, y_1) , and lower right corner (x_2, y_2) using the following equation:

$$\sum_{x=x_1}^{x_2} \sum_{y=y_1}^{y_2} f(x, y) = I(x_2, y_2) - I(x_2, y_1 - 1) - I(x_1 - 1, y_2) + I(x_1 - 1, y_1 - 1). \quad (3.5)$$

The grayscale pixel value in the original smoothed image is checked against the value generated by this sum and the output image pixel for a thresholded image is set to 0 or 1 depending on whether this sum is greater or smaller than the value of the corresponding pixel in the input image. This algorithm has proven effective for the images that we deal with.

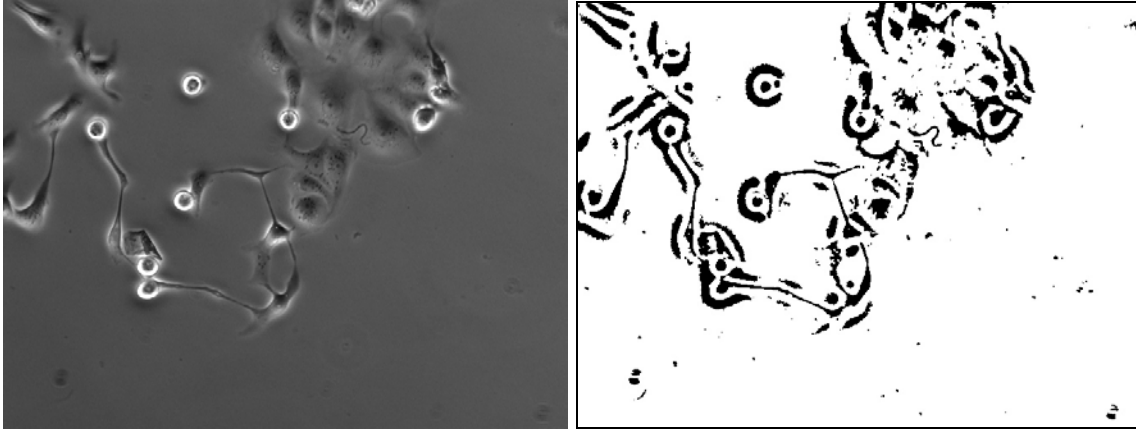


Figure 3-2 Left: Original image, Right: Image after applying adaptive threshold

To summarize the operation so far, given original two dimensional image frame f , binary thresholded image I_T is obtained as follows:

$$I_T = A(G(f)), \quad (3.6)$$

where A and G are the adaptive thresholding and Gaussian functions respectively. Figure 3-2 shows the effect of applying an adaptive threshold to an input image f on which Gaussian smoothing is applied. The cells and RFs appear as areas of black pixels, while the background appears as white pixels. Once we have thresholded the input image, our next step is to eliminate the regions corresponding to cell structures. The next section elaborates the step of discarding the cell structure regions.

3.4 Discarding cell structure regions from the binary image

In the binary image of Figure 3-2, we are interested only in the fibers. Therefore, we remove the black pixels corresponding to cell structures by eliminating the thicker regions and retaining the thinner sections of the foreground image. The final preprocessed image I_P can be obtained by applying the following function to I_T ,

$$I_p(x,y) = \begin{cases} 0 & \text{if } I_T(x,y) = 0 \text{ \& } \\ & I_T(x+i,y+j) = I_T(x-i,y-j) = 1 \\ & \text{for any } (i,j) \text{ where } 0 \leq i < 3 \text{ \& } 0 \leq j < 3 \\ 1 & \text{otherwise} \end{cases}, \quad (3.7)$$

where pixel value 0 stands for black and 1 for white.

The function operates in the following way: All the white pixels in adaptive thresholded image I_T are stored as white pixels in the output image I_p , at their corresponding locations. Each black pixel in I_T is examined to determine if it is surrounded by white pixels on any of its opposite sides in a 7×7 pixel window neighborhood. If there exists any such pair of white pixels then the black pixel is stored as a black pixel in the output image I_p else it is stored as white in the output image. An example pair of white pixels at opposite sides of a black pixel is shown in Figure 3-3.

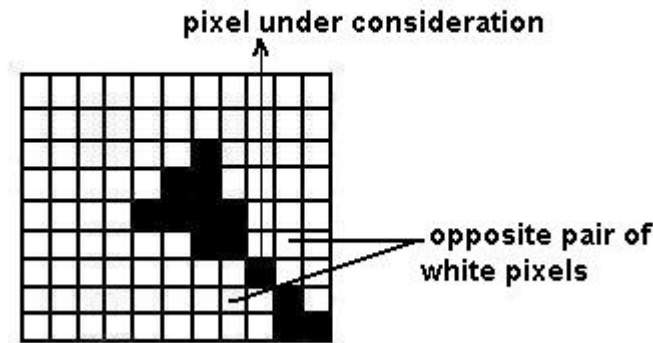


Figure 3-3 Processing example for maintaining thinner regions

For the black pixels corresponding to cells, such a pair of opposite white pixels is usually not found as the black area generally extends beyond the 7×7 area. Therefore these black pixels get stored as white pixels. In contrast, for the black pixels corresponding to fiber, one or more pairs of opposite white pixels is generally found, and therefore these black pixels get stored as black in

the output image. As a result we see in I_P black pixels corresponding to thinner black regions of I_T .

Figure 3-4 shows the preprocessed image I_P obtained from image I_T in Figure 3-2. Note the removal of most cell pixels at this step in processing. The long connected regions correspond to thin structures in the original image of Figure 3-2. The image areas corresponding to these thin structures can be used to identify RFs.

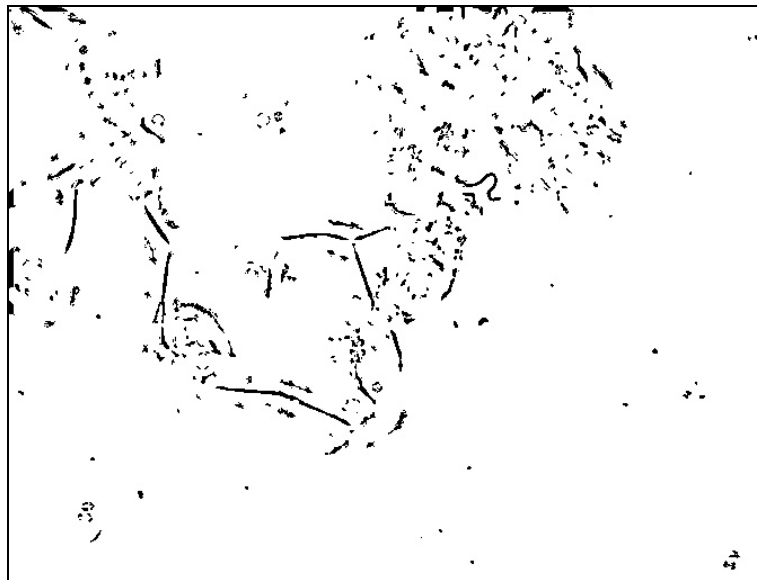


Figure 3-4 Binary image of Figure 3-2 with cell pixels removed

4 DETECTION AND TRACKING OF RFS

Image pre-processing produces images with features corresponding to thin long structures in the original image. The binary image obtained from pre-processing includes groups of black pixels that are background noise. We remove these undesired groups of black pixels by using a connected components algorithm. This chapter details the strategy used to remove noise regions with a size less than a threshold ρ , and it details the information that is stored for the desired image regions as well as details of detecting and tracking RFS.

4.1 Connected components

Connected components in an image are discrete regions of similar pixel intensity. In a binary image as shown in Figure 3-4, connected components are the discrete regions of black pixels. Not all these discrete regions represent *potential* locations of RFS, let alone *actual* locations of RFS. From chapter 1 we see that RFS are generally straight long structures stretched between cells. Hence, in this step, we filter the connected components that are considerably smaller in size (having fewer total pixels) using a threshold ρ . The ρ selection is very critical in this implementation. Selecting a high value for this threshold causes RFS that are short to be ignored, whereas selecting a low value causes many cell regions to be identified as potential RFS. After experimenting with different values, ρ was assigned the constant value 13 in this implementation. That is, connected components with at least 13 pixels are processed further to detect RFS.

In image processing terminology, an algorithm that finds connected components is called a *connected component labeling* algorithm. This algorithm scans the image having discrete regions and groups its pixels into components based on pixels that share similar intensity values and that are adjacent in the image. It further labels each component with a unique identity to distinguish the regions. For our purpose it is sufficient to identify the components to check the number of pixels in each. We do not label these components. In implementing the connected component labeling algorithm, we determined that it was not working efficiently and rather cumbersome for images with distinct regions very close to each other. Therefore, we resorted to using the `cvFindContours` function in OpenCV to find all the connected components [15]. To satisfy the function call pre-requisites, we first find the edges in the binary image (see Figure 3-4) using a Canny edge detector. The connected components are referred to as ‘contours’ by the function call, and each contour area is checked for number of pixels. As mentioned earlier, contour areas 13 pixels or larger are considered for further processing and analysis. We define I_R as the new image consisting of these contours. Figure 4-1 shows the contours of the regions in the resulting I_R given the I_P in Figure 3-4. As can be seen, it has less clutter than the original image in Figure 3-4.

4.2 RFs extraction

We assume that the counters extracted in earlier image processing steps serve as areas of potential RFs. In other words, these contours act as pointers to areas in the microscopic image that may contain one or more RFs. The contours returned by `cvFindContours` function are not themselves processed further; instead they provide the location and dimensions of the contours in the image as shown in Figure 4-1.



Figure 4-1 Contours of size 13

Recall that biologists define RFs as thin structures that are stretched between two cells, indicating regions of tension caused by cells pulling apart. These tension areas appear as lines in the microscopic images. Our strategy involves seeking ‘lines’ in the microscopic images at locations pointed out by the contours supplied by the above image processing step. Seeking lines in limited areas (only the contour areas) instead of over the entire image has two advantages. First, it is computationally inexpensive, and second, the dimensions of a line can be more easily determined.

While the structure of these RFs is well defined, the microscopic images do not always show sufficient contrast of RFs relative to the background, and it is therefore difficult to identify the lines reliably in all cases. Figure 4-2 summarizes the image processing operations done so far and indicates the steps taken towards RF extraction. (Processing steps not yet addressed will be discussed shortly.)

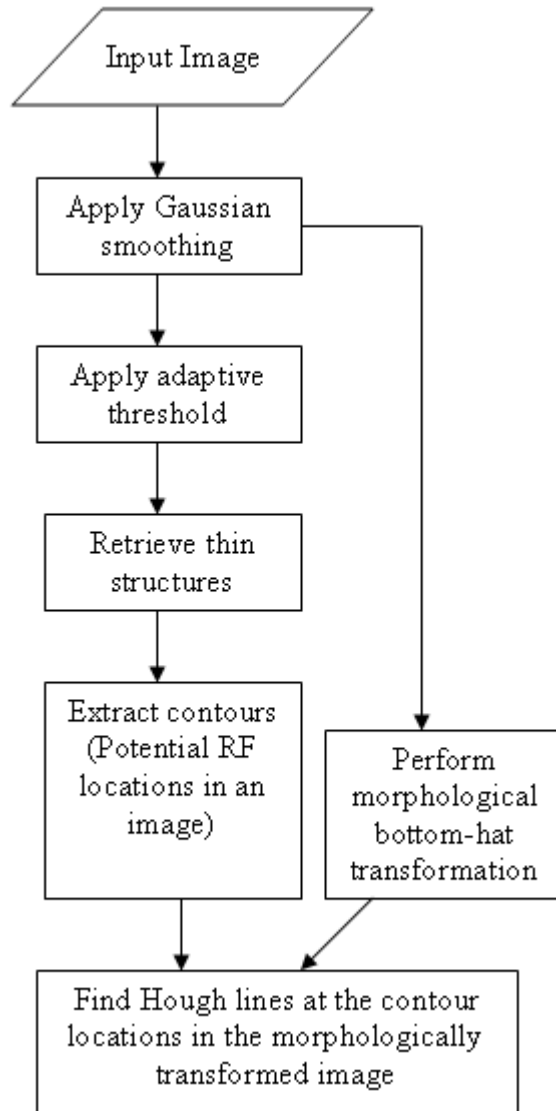


Figure 4-2 RFs extraction in an image

4.3 Black Hat transform

To extract RFs at the potential locations, we perform a morphological gradient on the original smoothed microscopic image using Black Hat Transformation [15], also referred to as bottom-hat transformation [17]. A morphological gradient is the difference between the morphological dilation and morphological erosion of a given image. Dilation is an image

processing operation to grow or thicken the foreground regions. Erosion is an image processing operation to shrink or thin the foreground regions. Various combinations of dilation and erosion operations are used to capture the contrast intensity in images as well as to isolate parts of an object that exhibit brightness changes relative to the object to which they are attached. One such combination is the bottom-hat transform operator. Bottom-hat transformation is often used to detect dark objects on a light background. It is represented mathematically as:

$$\text{Bottom hat (Image)} = (\text{Erosion (Dilation (Original Image))}) - \text{Original Image}, \quad (4.1)$$

where the original image undergoes dilation followed by erosion, and the resulting image is subtracted from the original image to give the bottom-hat transformed image.

In this application, we wish to detect darker objects in the image that include the RFs. The resulting image becomes

$$I_B = B_{hat}(G(f)), \quad (4.2)$$

where B_{hat} and G is the bottom-hat transformation and Gaussian function respectively, and f is the original image.

At this point of processing we have a list of contours which are shown for convenience and debugging purpose in the contour image I_R , and the morphologically transformed image I_B . Each contour in the list has its image location and rectangular dimensions determined. The rectangular section of image I_B corresponding to this information is sought, and in this sub-image lines are detected. The rectangular region of I_B may or may not have lines. The process of finding lines in I_B is repeated for all the contours in the list. The algorithm for line detection is detailed in next section describing the Hough transform.

Besides bottom-hat image transformation, we experimented with other image transformations in pursuit of better results for RF extraction:

- Edge image using Canny edge detector: In this case, we found the edges in the original microscopic images using Canny edge detection [14] with the observation that lines indicate abrupt changes in intensity and edges have similar characteristics. In the edge image we attempted to find lines in areas specified by the contours as described earlier. Using this approach it was possible to detect 50% of RFs in one of the available image sequences.
- Laplacian of an Image: The Laplacian operation in image processing is another well-studied and frequently applied edge detection mechanism. It provides the second-order derivative of the image. Edges are represented by the second-derivative values in the image where the zero-crossings appear. We used this approach to identify lines in edge images in the specified areas and it provided results similar to those provided by the bottom-hat transformation. Hence additional analysis was not done using this approach. However, Laplacian edge detection is used in post-processing for reasons that will be addressed.

We report results using the above two approaches for one of the available image sequences in the ‘Result’ chapter of this thesis.

4.4 Hough transform

The Hough transform [18] is a popular feature extraction technique used when simple shapes such as lines, circles or ellipses need to be detected in an image. It involves a voting procedure that uses a parametric description of the shape to detect instances of the shape in the image. The Hough transform is a vital part of the approach we propose for the detection of fibers. It was selected based on published studies that show its effectiveness in detecting simple

geometric shapes. In [19], Candamo et al. discuss the detection of thin lines in low quality videos. They propose an algorithm for use in low altitude aerial vehicles to detect obstacles such as cables, power lines and wires. Using the Hough transform for line detection to detect these obstacles, their work shows that the Hough transform is robust even when the contrast between object and background is low. Iisaka et al. [20] detect V-shaped features from SAR images, which generally suffer from low resolution. Line detection using the Hough transform was the core element of their process to detect V-shaped features. As a possible area of improvement, they suggest that a localized Hough transform would improve the efficiency of detection. Likewise, in our implementation we do not detect lines in the entire image plane but at certain candidate areas of the image, using the Hough line transform [14] [15]. Ren et al. [21] used the Hough transform for lane detection on the iPhone due to the transform's robustness to noise and occlusion, and also due to the method's simplicity and computational efficiency.

4.4.1 Hough line transform

We use an OpenCV Hough transform algorithm to detect lines [15]. The algorithm uses a parametric description of a line in polar coordinates (r, θ) which is represented as:

$$r = x \cos \theta + y \sin \theta, \tag{4.3}$$

where r is the distance of the line from the origin and θ is the angle formed by the normal of the line to the horizontal.

Local maxima in the parameter space (r, θ) , called the Hough accumulator space, reflect good candidates for lines in the image [17]. The OpenCV Hough transform function call therefore simply returns the local maxima in the (r, θ) space. The function call allows a user to

specify a threshold value that must be reached in Hough accumulator space to report a line. This threshold λ has been set to 20 pixels in our implementation.

As stated earlier, in each transformed image I_B , we use the Hough transform to find lines at every geometric location corresponding to contours in I_R . Lines with a length greater than a certain threshold λ are considered to be RFs when they are located within rectangles corresponding to bounding boxes for the regions in I_R . Figure 4-3 shows the RFs detected by this method given the contours of I_R in Figure 4-1.

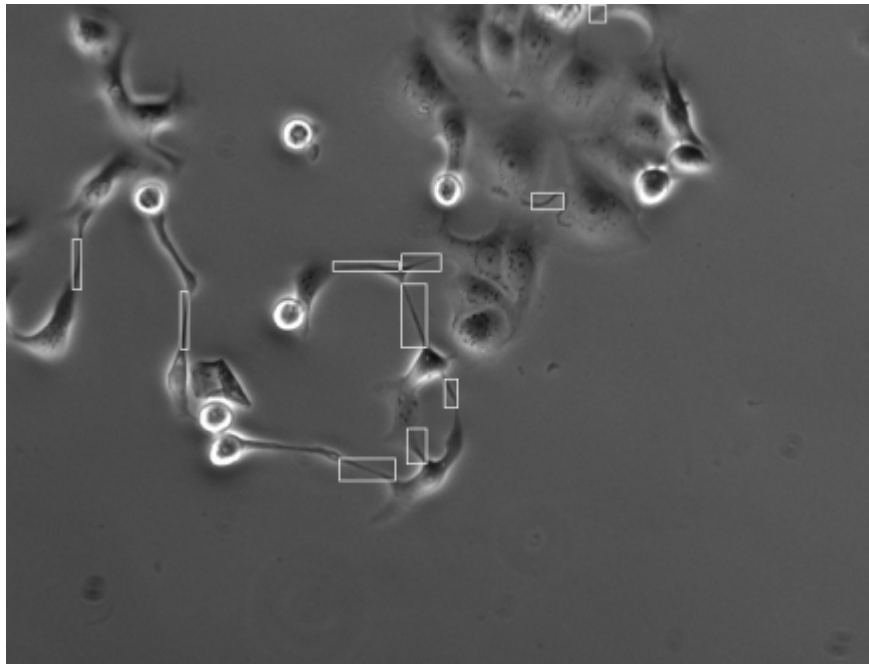


Figure 4-3 RF extraction

We store the RF statistics of each frame, including the number of fibers in each frame and the length and location of each. For example, in Figure 4-3, ten fibers are detected and their length and location information is stored. This serves as raw fiber information. The information is processed further to drop fibers that are false positives (FPs).

4.4.2 Post-processing of the detected lines

The above detection method provides a significant number of false positives in addition to correctly detected RFs. Therefore, we added a post-processing step to reduce the number of FPs. Figure 4-4 shows the strategy that reduces FPs.

According to this strategy, if a line is found at a contour location of the transformed image of I_B , the location is reprocessed to see if a line can be found in another transformation of the given image. The following steps are employed to achieve this purpose:

- The Laplacian operator is applied to the contour location to get an edge image.
- The edge image is thresholded using an iterative optimal thresholding technique, another variable thresholding technique described in Sonka et al. [14]
- The Hough line algorithm is applied to the thresholded image to check for lines, and line information is stored for tracking if a line is found.

Figure 4-5 shows an image frame in which eight RFs are detected in the bottom-hat transformed image. If the above strategy is employed, two of the RFs get dropped where a line could not be found. Figure 4-6 shows this effect. Thus, using this approach the number of false positives drop; however, this improvement comes at the cost of decreasing the detection of true RFs. Hence, there is a tradeoff between a higher RF detection with a higher number of false positives, and a lower RF detection with a lower number of false positives. If the false positive rate is uniform with the higher RF detection between different experiments (with different image sequences), then there is a possibility of providing useful information with a known degree of error. It will be shown in the Results chapter of this thesis that a careful selection of important threshold parameters such as line length plays a critical role in the RF detection rate.

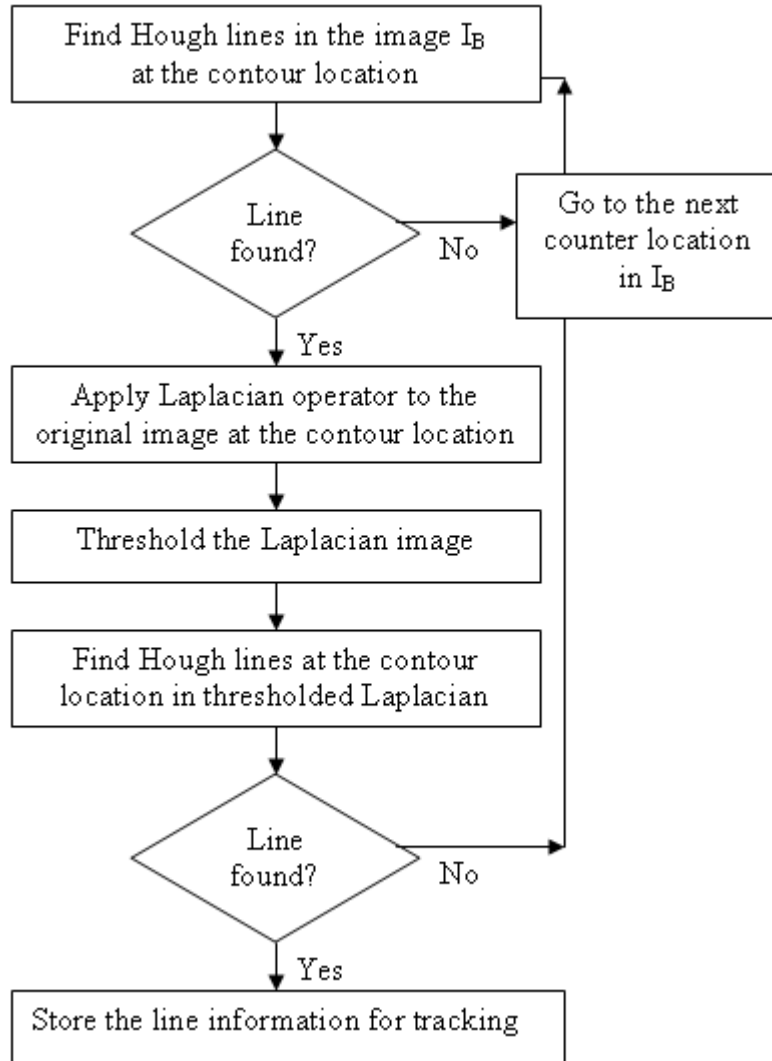


Figure 4-4 Strategy to reduce FPs

4.5 Tracking RFs

Using the RF statistics collected for all individual frames of the EMT image sequence, we track the number of unique RFs in the sequence by linking fibers in close proximity in adjacent frames. Each linked retraction fiber is given a globally unique identification number (fiber ID or Fiber ID) across the frames in the sequence.

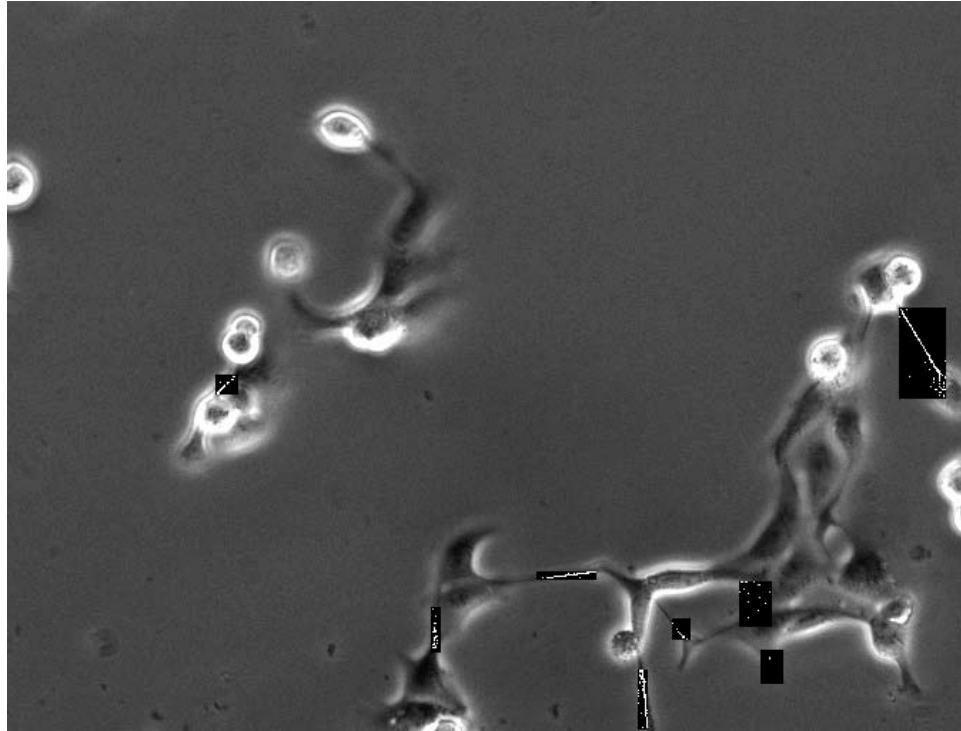


Figure 4-5 Eight RFs detected in bottom-hat image

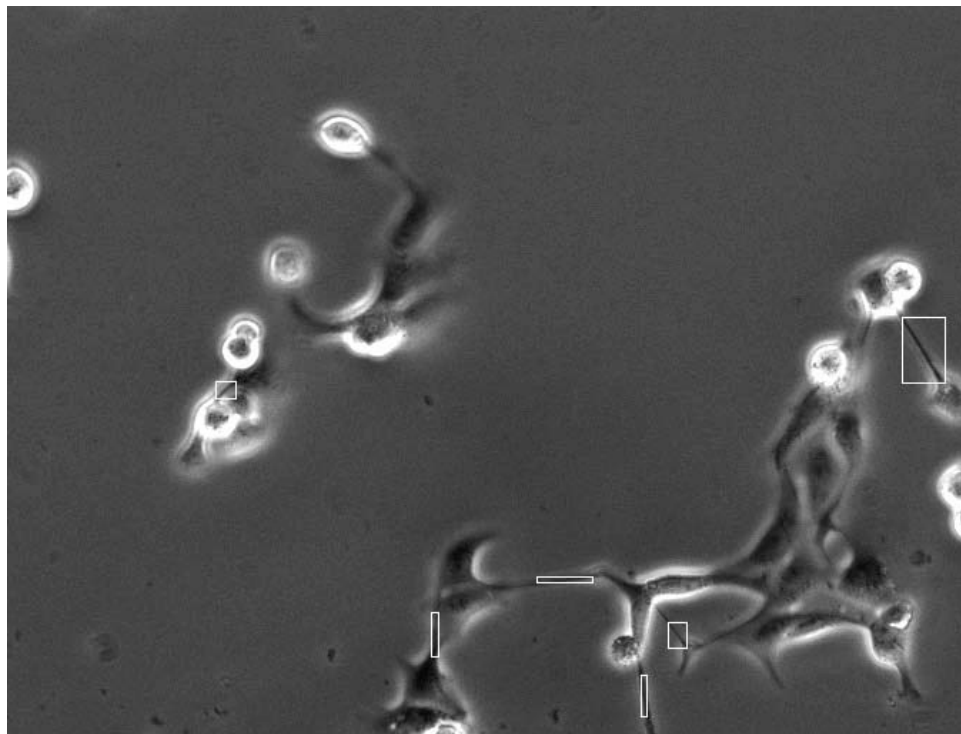


Figure 4-6 Six RFs detected as a result of post-processing step

Figure 4-7 illustrates RFs being tracked and linked across a three-frame sequence. At the end of all such processing for all frames, we obtain the total number of RFs present in the EMT image sequence, each of which has a unique fiber ID. For each retraction fiber, its lifetime is calculated by the number of frames in which it is present multiplied by the time interval between frames.

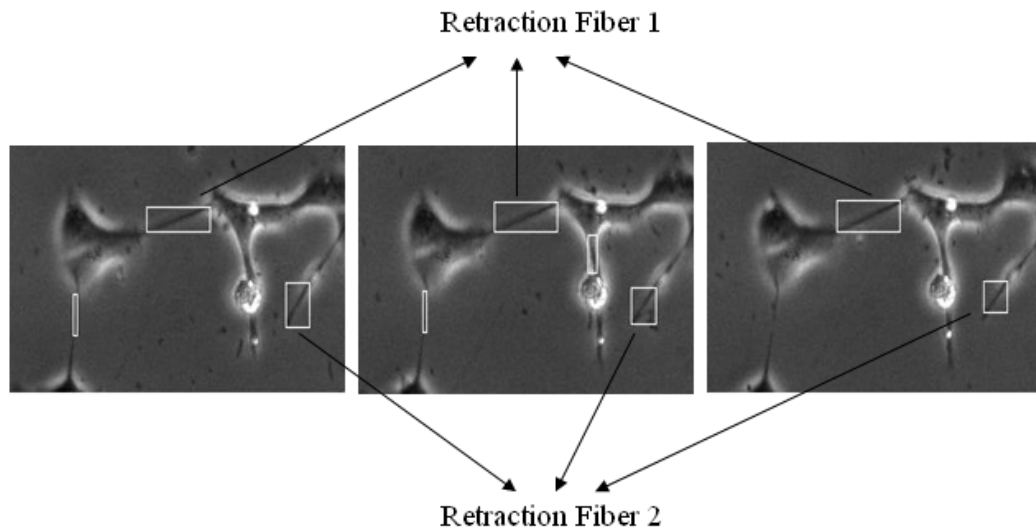


Figure 4-7 Example of two RFs tracked in a three-frame sequence

4.5.1 Tracking methodology

In the tracking phase, we link fibers in close proximity in adjacent frames using an approach involving overlapping rectangles. The rectangle used is the bounding box of a fiber in a frame (Figure 4-3). If a particular rectangle overlaps another rectangle in an adjacent frame at the same location, we assume that those two rectangles are correlated and the fibers they bound are the same retraction fiber. We perform this operation starting from the first frame, considering all the fibers in this frame one by one, and then proceeding to the next frame. If a fiber in an

adjacent frame is correlated, it is marked as processed, and is not considered for tracking again when the fibers of this adjacent frame are processed for tracking. Tracking this way, if a fiber exists for two or more frames, we provide it a unique fiber ID. In other words, if no correlation is found for a certain fiber in a frame, then that fiber is considered a false positive and is not further considered in the analysis.

If, for a particular rectangle, we do not find an overlapping rectangle in three consecutive frames, then we terminate the tracking of that particular fiber. This termination coincides with the end of the fiber's lifetime.

4.6 Detection and tracking example

Figure 4-8 shows an excerpt from program output files that reflect detection and tracking of RFs as described above. The left portion of the figure indicates the number of RFs detected in individual image frames of an example 20-frame sequence, and the right portion of the figure indicates the tracking of RFs across these image frames. Detection information suggests that the image frame 181 includes five detected RFs, while image frame 182 has seven RFs, and so on. The columns *X-coor* and *Y-coor* indicate spatial coordinates of the retraction fiber (line). The *Length* indicates an approximate length of the fiber. Note that the length information is very sensitive to illumination, pre-processing operations, and the transformed image on which line detection is performed. Therefore, in this study the length information does not reflect accurate information.

The tracking on the right side of the figure illustrates that fiber 1 exists from image frame 181 through image frame 186. Similarly, fiber 2 extends from image frame 181 through 184. Retraction fiber lifetime is calculated by determining the difference between the first frame and

the last frame in which it is present multiplied by the time interval between frames. Therefore, if we assume that the time interval between frames is two minutes, then the lifetime of fiber 2 is calculated as $((184-181)+1) * 2 = 8$ minutes.

Image No.	X-coor	Y-coor	Length	Fiber No.	X-coor	Y-coor	Length
Image 181				Fiber 1			
1	294	459	18.3848	181	294	459	18.3848
2	597	454	24.1661	182	295	460	20.1246
3	333	395	60.8276	184	296	463	19.4165
4	642	243	20.2485	186	300	465	21.4709
5	38	203	37.1618	Fiber 2			
Image 182				181	597	454	24.1661
1	295	460	20.1246	182	588	439	40.7063
2	588	439	40.7063	184	592	436	33.541
3	351	401	28.6356	Fiber 3			
4	36	201	39.9249	181	333	395	60.8276
5	464	442	23.0868	182	351	401	28.6356
6	442	439	60.1332	183	351	398	56.8858
7	232	89	27.5862	184	357	403	21.2132
Image 183				186	352	398	47.0744
1	351	398	56.8858	187	369	397	33.7343
2	650	246	27.2947	189	346	403	30.5941
3	34	203	31.6228	190	357	398	46.6905
4	445	452	23	191	359	401	26.1725
6	464	442	23.0868	192	371	400	39.2046
7	232	89	27.5862	195	386	402	25.0799
Image 184				Fiber 4			
1	296	463	19.4165	181	642	243	20.2485
2	592	436	33.541	183	650	246	27.2947
3	357	403	21.2132	185	646	235	40.2492
.				.			
.				.			
A. Detection				B. Tracking			

Figure 4-8 RFs detection and tracking

5 RESULTS

This chapter discusses the performance of our system in the automatic identification and tracking of RFs. Our system is implemented under Microsoft Windows in C using image processing functions provided in the OpenCV library [22]. Three different EMT image sequences were provided by the Cancer Research Center in the Department of Physiology and Developmental Biology at Brigham Young University. Representative windows of frames in each of the three sequences were selected and processed with the proposed method. Each image sequence was reviewed by biologists, who manually identified the fibers and their lengths in each image. The statistics provided by the biologists are used as ground truth to evaluate our results.

As noted earlier, the system effectiveness – maintaining a high RF detection rate – depends on multiple factors. These include the quality of the image sequence, performance of adaptive thresholding, the ρ -threshold used for connected component algorithm to eliminate noise, the edge image for Hough line detection, the λ -threshold value to be reached in the Hough accumulator plane to be reported as a line, and the performance of post-processing. To evaluate our system effectiveness, we define the terminology to be used as follows:

For a given image sequence, let x be the number of unique RFs identified in the benchmark across the image sequence, and n be the number of unique RFs (fiber IDs) detected automatically by our method for the same. Each of these n identified fibers can be classified within one of the following categories:

- True Positives (TP): The number of actual, unique RFs that are correctly detected
- False Positive (FP): The number of objects that are not RFs but incorrectly identified as such
- Multiple Identities (MI): In this implementation, there are certain “true” RFs that do not have one to one correspondence between them and the detected fiber IDs. In other words, a single “true” fiber is identified as multiple distinct RFs over time. Such a true fiber is said to have multiple identities. Therefore, MI represents the total number of RFs occurring in multiple identities of TP while tracking.

Furthermore, those RFs identified manually in the benchmark (part of the ground truth) which are not detected by the system are referred to as false negatives (FN).

For example, consider an image sequence of 100 frames and let x be 20 and n be 30. The 30 detected fibers are the sum of TP, FP and MI. Let us say that the system has correctly identified 15 RFs, and that 13 of the identified fibers are not actually RFs. Then TP is 15, FP is 13, and we can conclude that MI is 2, meaning that two detected fibers correspond to one or more detected true positives. That is, there is a possibility that one true fiber of the example is detected by 3 RFs (one considered in TP and 2 in MI), but it is also possible that there are 2 “true” fibers, each of which corresponds to 2 detected RFs (one considered in TP and one in MI). Since 5 true RFs are not detected in this example, FN is 5.

The *RF detection rate* is calculated as the ratio of TP to x . It is also referred to as the statistical *overall sensitivity* $S_{overall}$ of the automatically generated output. The RF detection rate for the above example is $15/20 = 75\%$.

We also define the *FP detection rate* as follows. From the tracking results (see right side of Figure 4-8), we first determine the correctly detected fiber IDs of the sequence (TP and MI),

and sum up the detected fibers for each of these fiber IDs. False positives are then computed as the difference between this sum and the total detected fibers in the tracking results. The *FP detection rate* is the ratio of these false positives to the total detected fibers. To illustrate, let us assume that n , the total number of actual fibers (one instance per frame) over a given image sequence, is 1000, and that the sum of detected fibers matching the actual fibers is 600. Then the *FP detection rate* = $(1000-600)/1000 = 40\%$. It is highly desirable to have a high RF detection rate and low FP detection rate.

The following sections present the results of various experiments in which algorithms and parameter values are modified and the resulting performance are reported. The bottom-hat approach is employed for the results unless otherwise specified.

5.1 Edge image for line detection

We conducted experiments with three different approaches, bottom-hat, Laplacian, and Canny. The bottom-hat column in Table 5-1 indicates our main approach that is discussed in this thesis work. We slightly modified this approach by replacing the bottom-hat edge image with the Laplacian edge image and recorded the results. In the next variation of our approach, we replaced the bottom-hat image with a Canny edge image and noted the results. The number of RFs detected over the given image sequences varied using the three approaches, as shown in Table 5-1 for one of the image sequences. The image sequence considered for this experiment has 160 image frames, and x , the number of unique RFs identified, is 48.

As can be seen from the table, the maximum number of RFs was detected using the bottom-hat approach. Further, it is important to note that the 30 RFs detected in the image sequence by the Laplacian are a subset of the 32 RFs detected by the bottom-hat approach. This

is true with the Canny approach as well, where the 27 RFs detected are a subset of the 32 RFs detected by bottom-hat. Therefore, all results that follow employ the bottom-hat approach, unless otherwise specified. If the FP detection rate does not vary significantly between different experiments (image sequences), then we can consider using the bottom-hat approach to estimate the number of RFs formed during EMT. However, if it varies significantly between sequences, then the system will fail to provide the necessary accuracy. The Canny approach provides the lowest FP detection rate, but it is accompanied by a failure to detect certain RFs. The RFs detected by Laplacian approach are the same as those detected by bottom-hat with a loss of just two RFs for the image sequence considered.

Table 5-1 RFs detection and tracking using the 3 approaches

	<i>Bottom-hat</i>	<i>Laplacian</i>	<i>Canny</i>
<i>n</i>	137	138	74
TP	32	30	27
MI	4	3	4
<i>S_{overall}</i>	66.7%	62.5%	56%
FP detection rate	62.35%	65.3%	56.2%

Table 5-1 also shows the FP detection rate. This rate is high for the following reasons:

- Our approach detects RFs at places wherever it finds darker pixels that satisfy the λ -threshold. There are many locations in the cell colonies that are dark and thin that get detected and tracked, and hence difficult to delineate.
- During the EMT process, as cells tend to separate from each other, the cell membranes stretch and become darker with respect to background. Sometimes the entire cell changes shape and gets elongated such that it resembles an RF by being

thin, straight and dark. These are not straight RFs, but there are enough pixels to satisfy the λ -threshold and get incorrectly detected as RFs.

The results presented in Table 5-1 were obtained using the overall approach, including the steps of both identification and tracking. To know the detection accuracy of computer vision algorithms in the detection phase, we took a representative window of 11 frames from the image sequence and determined the precision. This gives an idea of the number of false positives detected in each of the frames and hence the detection accuracy. We discuss the precision of the representative window in the following paragraph.

We define precision P for individual frame as the ratio of the sum of TP and MI to N, where N is the total number of RFs detected in an individual frame; note that TP and MI represent true RFs in a single frame in this context. Figure 5-1 shows the precision of our approach in individual frames within the selected 11-frame window from image sequence 1. It can be seen from the graph that the detected RFs are mostly true RFs. The average precision for this window is 61.1% which indicates that the false positives are about 39%. This graph illustrates RF detection at the frame level. Time constraints prevented the computation of results over a larger window from this sequence, or computing results over windows of the other two image sequences.

The graph of Figure 5-1 reveals that FP detection is not as unsatisfactory as the FP detection rate shown by Table 5-1 if frames that are representative of RFs are considered. During the EMT process the RFs do not start to appear immediately. If the biologists could provide input as to when in the image sequence the RFs start forming in a significant way, then the initial frames of the sequence could receive special processing, which would in turn result in the elimination of a significant number of FPs in the overall image sequence.

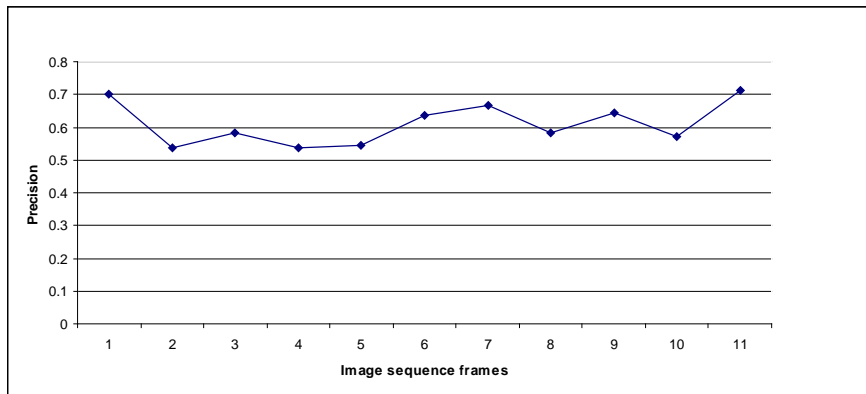


Figure 5-1 Precision of RF detection in individual frames of an image sequence

5.2 Effect of post processing

An option of post processing is employed in this implementation as described in Section 4.4.2. Table 5-2 summarizes the effect of the post processing strategy. Column 2 indicates the results of the bottom-hat approach that we discussed in Table 5-1 above, and column 3 shows the result of the approach if the optional strategy is employed. While we could reduce the FP detection rate across the image sequence using this optional step by 22.5%, it is accompanied by the undesirable effect of lowering the overall sensitivity of the system by 31.3%. Therefore, although the post-processing strategy has been discussed as part of this thesis work, it does not appear to be an advantage to employ it, particularly since biologists indicate that they would prefer a high overall sensitivity to a low FP detection rate.

Table 5-2 Effect of post processing strategy for RFs

	<i>Without post processing</i>	<i>With post processing</i>
<i>n</i>	137	40
TP	32	22
<i>S_{overall}</i>	66.7%	45.8%
FP detection rate	62.35%	48.3%

5.3 λ selection

The λ -threshold that is used to report a line by Hough transform is a critical parameter determining performance. As stated earlier, the value of this parameter was chosen to be 20, meaning that at least 20 pixels must satisfy the line equation given by equation 4.3 to be detected as a retraction fiber. Figure 5-2 and Figure 5-3 illustrate the effect of setting this threshold value to 20 and 15 respectively for an image frame transformed using the bottom-hat approach. While eight RFs are detected in Figure 5-2, 14 are detected for the same image frame in Figure 5-3. It can be seen that number of false positives increased from two in the first figure to eight in the later figure. Thus, while selecting a low value for λ increases the likelihood of detecting smaller fibers, it also increases the false positives to a significant level.

For one of the image sequences, where the number of true fibers was 48, using a threshold of 15 resulted in the detection of 257 fiber IDs, where 186 were false positives and the remaining 71 fibers reflected the detection of 32 true fibers using multiple identities. High MI values are an additional undesired effect of using lower threshold values. Apart from a significant number of false positives, lower thresholds may also reduce the accuracy of tracking. This occurs because multiple small fibers may get detected in areas close to true fibers and the overlapping bounding boxes tend to contribute to increased tracking errors.

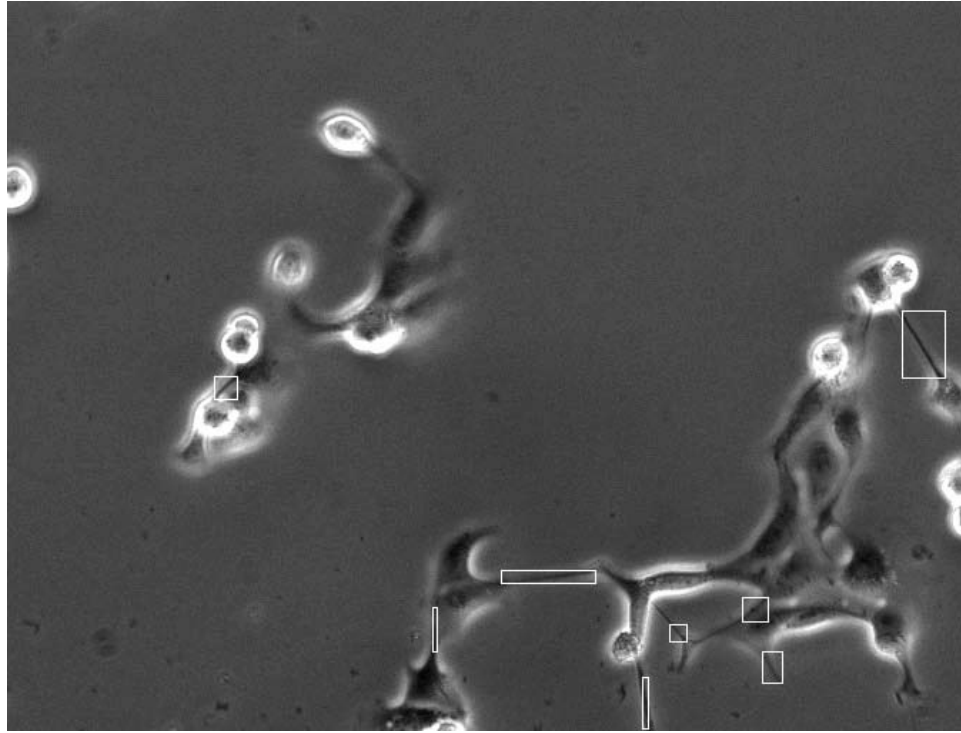


Figure 5-2 RFs detected with line threshold 20

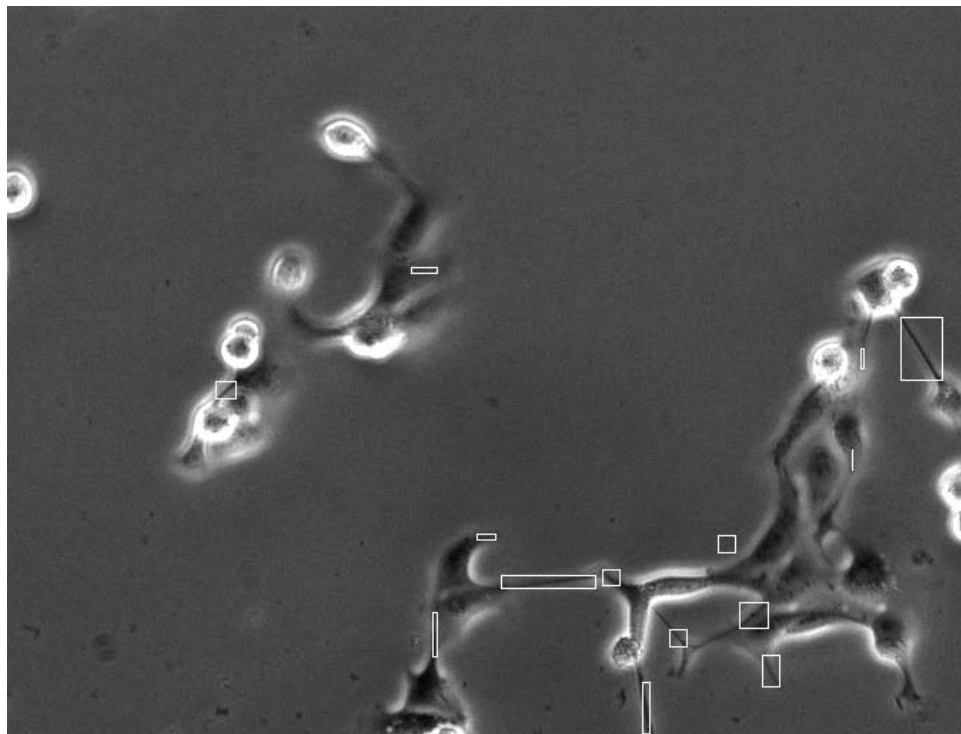


Figure 5-3 RFs detected with line threshold 15

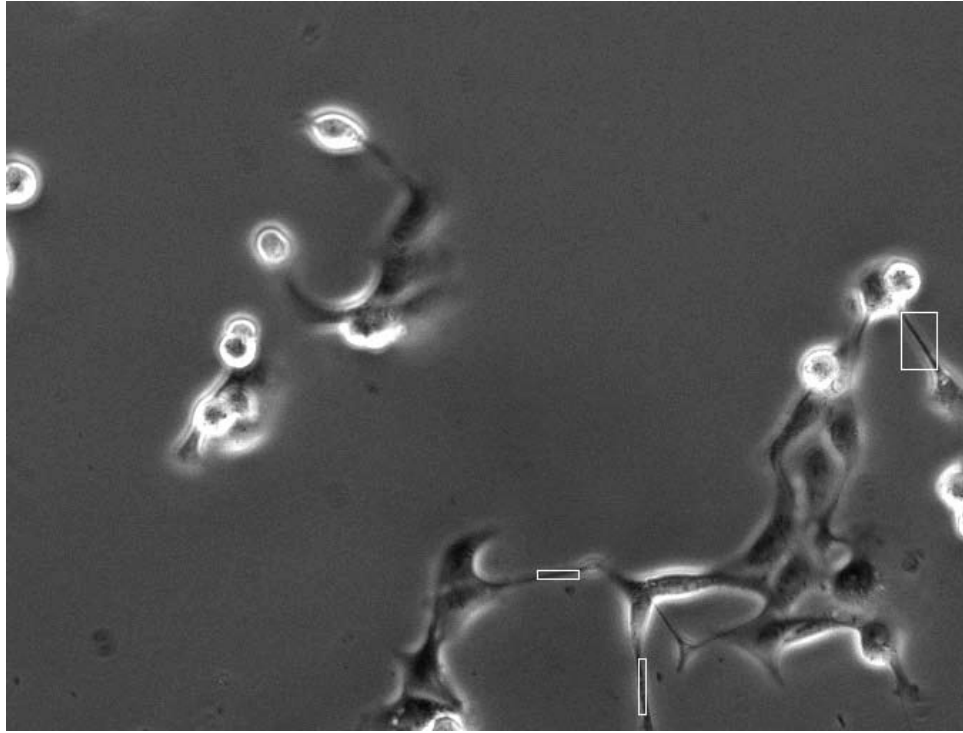


Figure 5-4 RFs detected with line threshold 20 in Laplacian transformed image

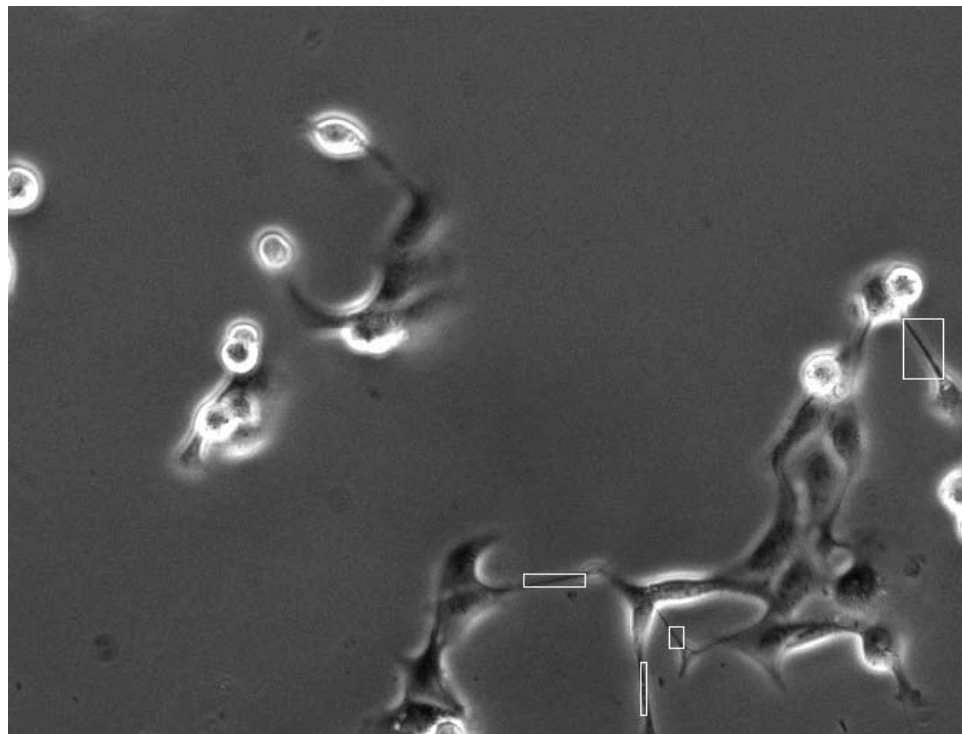


Figure 5-5 RFs detected with line threshold 15 in Laplacian threshold image

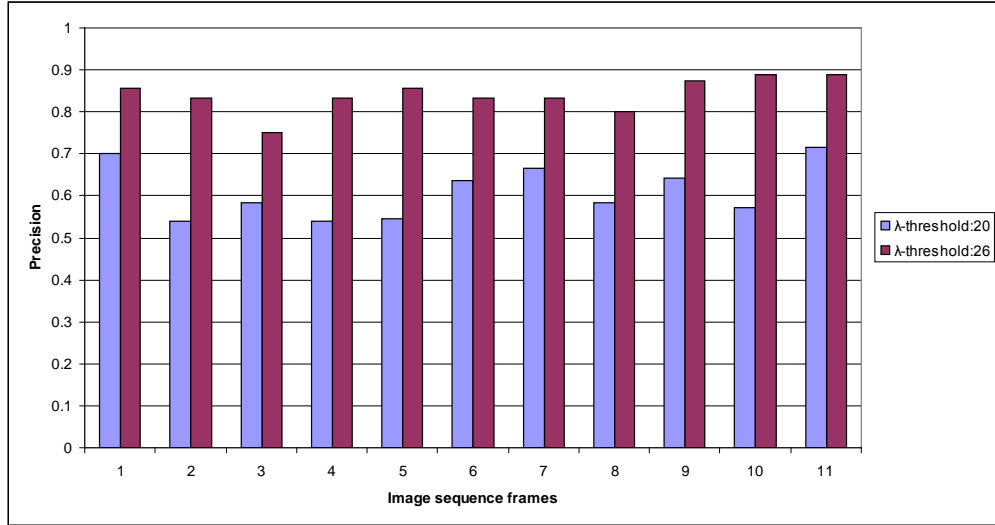


Figure 5-6 Effect of line threshold on precision

The λ effect can also be seen in Figure 5-4 and Figure 5-5, which employ the Laplacian approach. In this example frame, using a low threshold is beneficial as the system is able to detect one extra true fiber using a low threshold. Another example of the λ effect is found in the precision graph of Figure 5-6. The results show that using a higher λ -threshold improves precision.

5.4 System performance across different experiments

So far, in the above sections all experiments used the same image sequence to illustrate various significant factors affecting system performance. We need to evaluate our approach using the other two available EMT image sequences. Table 5-3 shows the system performance for the three image sequences, obtained using identical algorithms and parameter values.

Table 5-3 indicates that the performance is best for the second image sequence and worst for the third image sequence. In image sequence 3, contrast in the frames was not sufficient for the adaptive threshold to retain the image regions corresponding to ten RF areas. Figure 5-7

shows an example of how one RF was lost in adaptive thresholding and thus could not be detected by the system. The thresholded image (B) has no content corresponding to the fiber area. Almost ten RFs are lost due to two factors impacting either individually or in combination. These factors are poor illumination near an RF, and a short fiber length that failed to satisfy the threshold λ required to be considered an RF.

Table 5-3 System performance for the three image sequences

	<i>Image Sequence 1</i>	<i>Image Sequence 2</i>	<i>Image Sequence 3</i>
Number of frames in the sequence	160	140	120
n	137	128	189
x	48	40	44
TP	32	29	24
$S_{overall}$	66.7%	72.5%	54.5%
FP detection rate	62.35%	63.8%	81.7%

While the other two image sequences also suffered from poor illumination, low contrast, and the limited length for certain fibers, the results were better because there were a significant number of fibers whose life spans were long enough to allow the system to detect them. In image sequence 3, 35% of the 20 undetected fibers had life spans of 5 frames or less. Moreover, many RFs do not stretch enough to reach the threshold λ . This effect is less severe in the other two sequences.

5.5 Tracking results

Figure 5-8 shows examples of successful tracking using the bounding box approach in a window of six frames, in cropped spatial dimensions of image sequence 1 (upper) and image sequence 2 (lower) respectively. In the upper portion of the figure, two RFs are tracked. The

termination of one of the fibers at the far right is accompanied by the absence of its accompanying bounding box. Likewise, the lower cropped image frames show tracking of one RF using bounding boxes. The common feature in the accurate tracking of these RFs over all images in the figure is that the fibers formed are at a significant distance from one another, resulting in the correct assignment of a unique ID to each fiber. The third row of Table 5-4 shows the number of true fibers in each image sequence that are tracked using a unique ID.

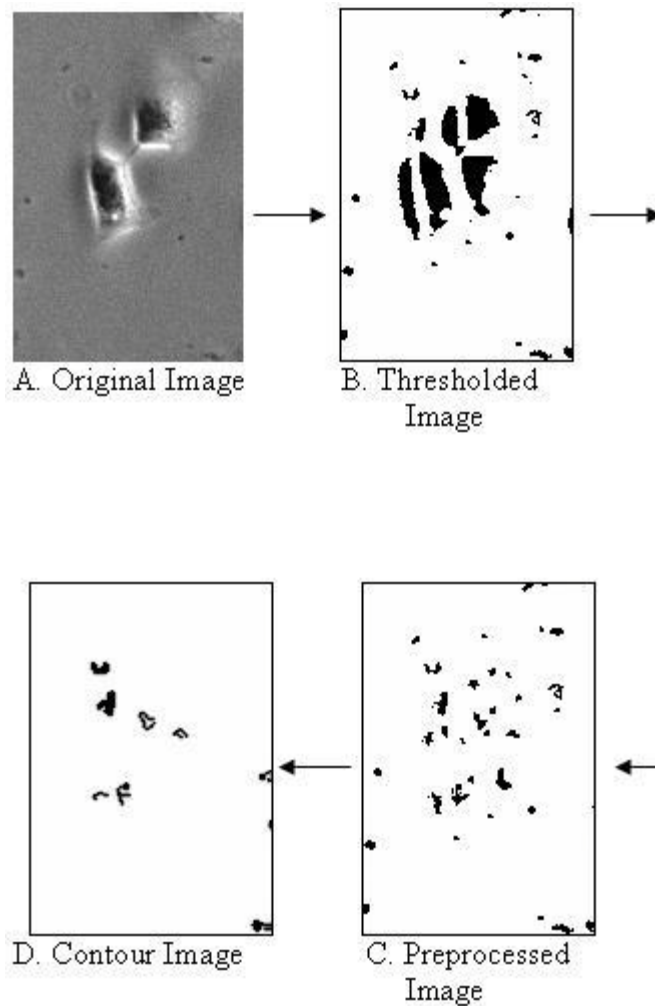


Figure 5-7 Example of adaptive thresholding contributing to missing RFs

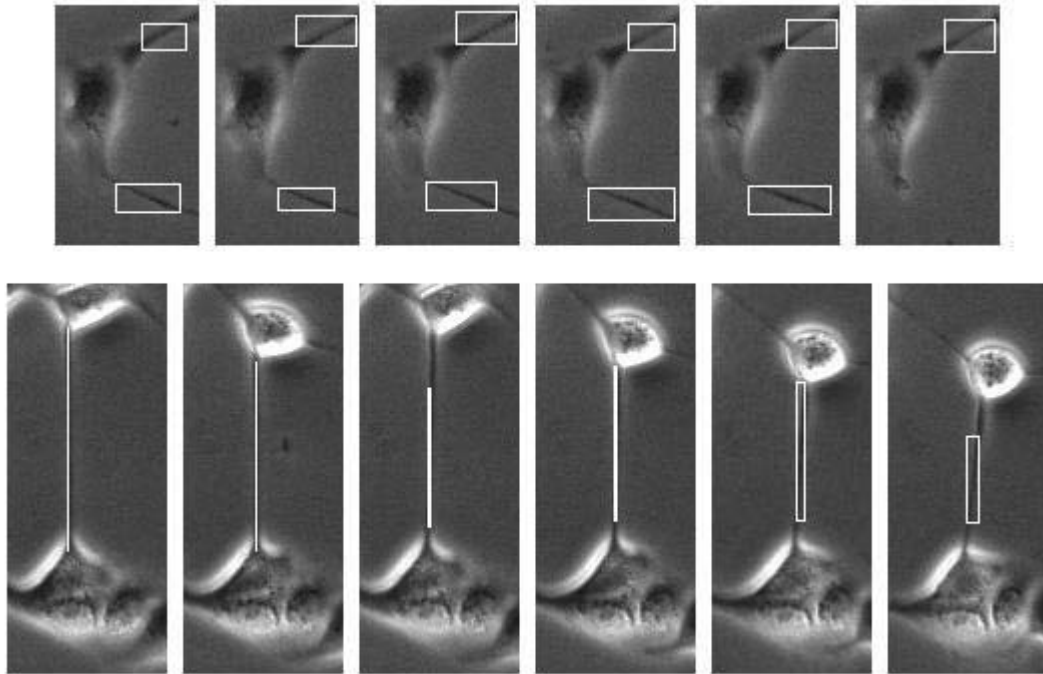


Figure 5-8 Examples of overlapping rectangles providing unique IDs for RFs

Table 5-4 Number of RFs tracked with a unique/multiple ID

	<i>Image Sequence 1</i>	<i>Image Sequence 2</i>	<i>Image Sequence 3</i>
TP	32	29	24
RFs with unique ID	28	20	20
RFs with multiple ID	4	9	4

The bounding box approach works reasonably well if not hampered by any of the following conditions:

- A single RF is detected at a different spatial location.
- RFs are formed in close proximity, thereby making it difficult to distinguish between them.
- RFs and false positives are detected in close proximity.

Figure 5-9 shows two example cropped images that illustrate the first problem stated above. In the upper portion of the figure, for the single RF the system generates two fiber IDs. In such a case we say that the RF has been identified by multiple IDs. Similarly, in the lower portion of the figure, the right fiber has been identified by multiple IDs. Further, in this case if a line is identified for one fiber and the adjacent fiber's bounding box overlaps, then the adjacent fiber information is not stored during the fiber identification stage. In the next frame, the adjacent fiber may be identified first and stored, and the former fiber may not be stored. This may result in ineffective tracking (not shown here), further contributing to the second condition mentioned above.

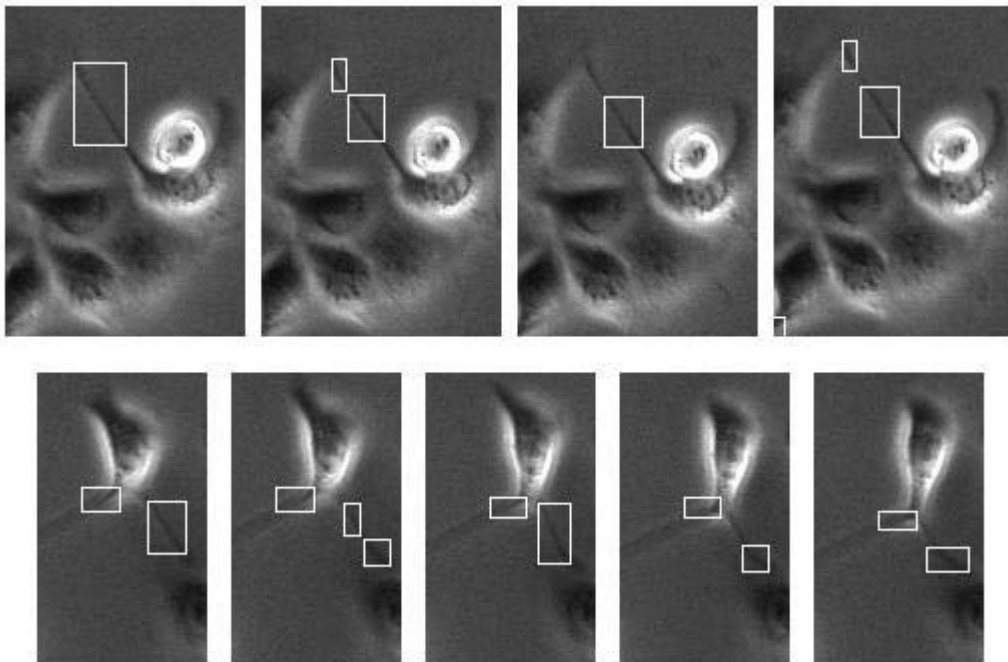


Figure 5-9 Examples of overlapping rectangles providing multiple IDs for RFs

The fourth row of Table 5-4 shows the number of RFs that used multiple IDs in each of the image sequences. This number is determined by the contour information available during the

RF detection stage. If the RF considered does not have uniform grayscale value then there is possibility of the single true fiber get broken into different contours. A line may be detected in each of the broken contours, thereby contributing to multiple IDs for a single RF. Broken contours may also result if the contrast available in the spatial neighborhood of a fiber is very low. The results in the table show that the undesirable phenomenon of having multiple IDs is more severe in image sequence 2 for the reasons stated.

5.6 RFs lifetime information

Table 5-5 shows the lifetime information of RFs formed during EMT. To determine how much the system lifetime information of RFs deviates from true lifetime information, we summed up the lifetimes of all the RFs detected by our system and compared it with the sum of the lifetimes of the RFs in the ground truth data. For example, our system detects 32 RFs in image sequence 1. Therefore, we summed up the lifetimes of these 32 RFs from the ground truth and compared it with the sum of the lifetimes of the 32 RFs detected by our system. We then calculated the average lifetime of the (true) benchmark RFs and system detected RFs by taking the respective sum and dividing it by the number of system detected RFs.

Table 5-5 Lifetime information statistics of RFs

	<i>Total lifetime of benchmark RFs</i>	<i>Average lifetime of benchmark RFs</i>	<i>Total lifetime of system detected RFs</i>	<i>Average lifetime of detected RFs</i>
Image Sequence 1	945	29.5	562	17.6
Image Sequence 2	834	28.8	464	16
Image Sequence 3	358	15	301	12.5

As seen from Table 5-5 the average lifetime of the RFs is shorter than the benchmark RFs. The reason for this reduced lifetime is because most of the RFs could not be tracked until the end of their lifetime. Here again, the dominating factor hampering lifetime information is the contrast in the spatial neighborhood of the RF. Figure 5-10 illustrates this effect. The fiber on the left side in the first frame gets tracked to a point where there is enough contrast for the system to pick up the line information for the fiber. However, from the third frame onwards, the system could not track the fiber and this led to an incorrect measurement of its lifetime. Another example of reduced contrast can be seen in the first frame of the figure. Ideally, the first frame should have shown two RFs. There exists another RF on the right hand side of the first frame that actually disappears in the next frame. However, the system terminated its tracking before it actually disappeared.

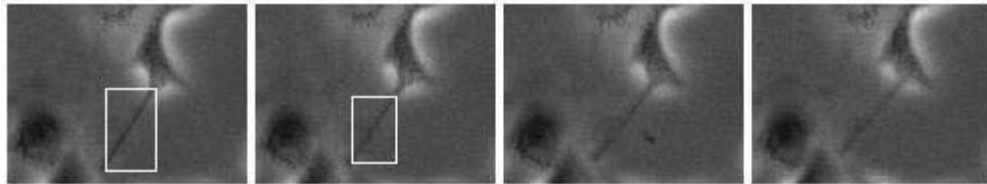


Figure 5-10 Example tracking of reduced lifetime of RFs

6 CONCLUSION

We have presented a method which shows potential for automatic identification of RFs during EMT in time-lapse microscopic image sequences. Our system was able to detect 72.5% of RFs for one of the three image sequences available. In the worst case, RF detection was 54.5% for another image sequence. In addition to identifying RFs, we also provided results for average lifetime information of the RFs in each of the test image sequences and discussed the reasons for the reduced accuracy lifetime statistics.

6.1 Identification of RFs

In this thesis work, we explored three alternative approaches for identification of RFs: bottom-hat transformation, Laplacian and Canny edge detection. We included results for each of these approaches for one of the image sequences (see Table 5-1). Further, we experimented with a post processing strategy for the candidate RF areas to reduce the number of false positives. The post processing results reveal that reducing false positive also reduces the number of true RFs detected (see Table 5-2). Hence, we conclude that achieving a high RF detection rate while maintaining a low FP detection rate is a difficult problem.

6.2 Tracking of RFs

For tracking, correlation between RFs in adjacent frames was achieved using overlapping rectangles that bounded the RFs. This strategy was computationally inexpensive and relatively

straightforward to implement. Problems of RF identity and ambiguity occur only when more than one RF moves to the same position or in close proximity in the adjacent frame. However, the effect is negligible in all the three image sequences that we studied.

6.3 Future work

Our approach utilizes digital image processing algorithms which are well studied and widely implemented. Therefore, taking this thesis as background work, further effort could be expended to find variations in the algorithm or algorithm parameters to see if the system performance can be improved. For example, we used the Hough transform implementation of OpenCV. Computer vision literature describes many variations of this algorithm for the voting procedure for detecting desired shapes. These variations could be explored to determine if the identification accuracy could be improved. Another example is the adaptive thresholding technique that we employed in our approach to address the challenge of variable illumination. It provided good segmentation results, but a significant number of fibers are missed because relevant information is lost in the adaptive thresholding step due to low contrast (see Figure 6-1). Hence, the thresholding technique could be explored further or perhaps replaced with more advanced techniques that could capture the fiber areas which are not captured by the current adaptive thresholding mechanism. Image contrast enhancement techniques such as histogram equalization could also be studied to find out if the low contrast near the retraction fiber can be enhanced.

Currently, during the identification phase, our approach assumes that in a candidate area there can be just one fiber, and we reserve this area for the fiber using a rectangle to exclude the rectangular area for further fiber identification. In other words, if two RFs are in the same

rectangular region, only one of these RFs is recorded during identification (depending on which contour is processed first). However, there can be another fiber in the same rectangular region. Hence, better fiber reserving technique can be utilized. The identification stage can be modified to allow multiple fibers under a threshold of two or three lines that can be considered in a rectangular region. And subsequently, the tracking approach needs to consider multiple fibers that may appear within the same region. This may be done by considering additional statistics such as the slope of the line, or gray-level statistics of line boundaries that may help distinguish multiple lines in the same rectangular region.

One of the steps that can be taken to reduce false positives in RF detection stage is measuring the RF width so that only very thin lines are accepted as RFs. In the tracking stage, the problem of multiple identities can be remedied by using information from consecutive frames to connect the broken segments or to enhance the contrast locally.

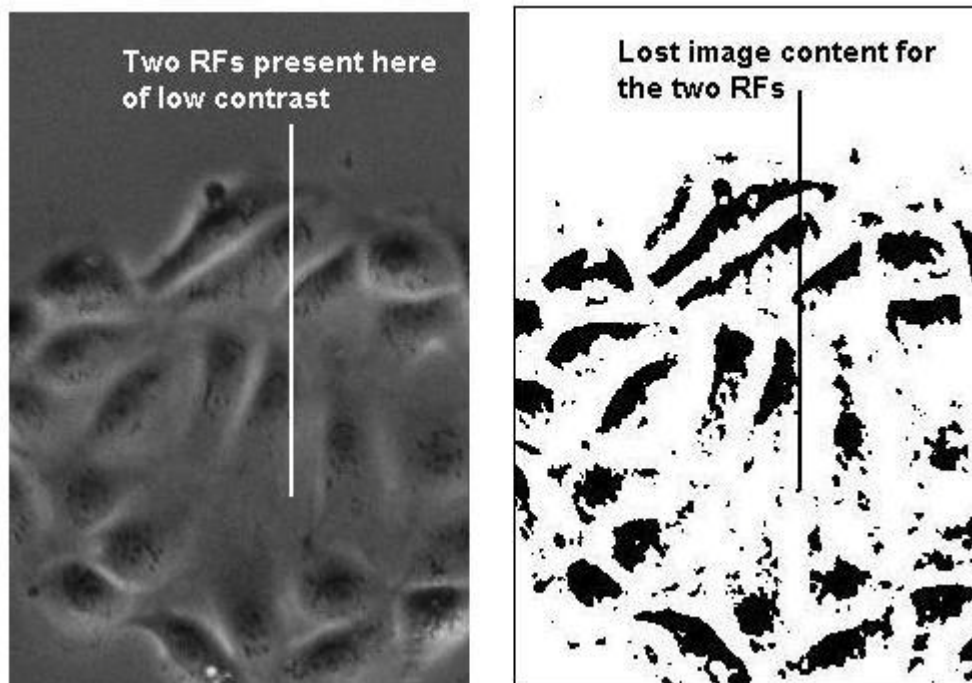


Figure 6-1 RFs lost at adaptive thresholding stage

REFERENCES

- [1] J. P. Thiery, “Epithelial-mesenchymal transitions in tumor progression”, *Nat Rev Cancer*, 2:442-54, 2002.
- [2] L. Sperry, N. Bishop, S. Maxfield, Z. Lewis, D. Staley, R. Vellinga, and M. Hansen, “Zyxin Associates with Dynamic Actin Structures and Alters Cell Scattering Response in HGF-Treated MDCK Cells”, *American Society for Cell Biology*, 2008, Dec. 13-17. Poster #2525/B234.
- [3] M. D. Hansen and M. C. Beckerle, “Opposing roles of zyxin/LPP ACTA repeats and the LIM domain region in cell-cell adhesion”, *J Biol Chem* 281, pp. 16178-16188, 2006.
- [4] http://www.olympusamerica.com/seg_section/product.asp?product=1054. Accessed 4/7/2009.
- [5] K. Li, E. D. Miller, M. Chen, T. Kanade, L. Weiss, and P. Campbell, “Computer vision tracking of stemness”, in *Proc. of IEEE International Symposium on Biomedical Imaging (ISBI): Special Session on In Vivo Microscopic Image Analysis*, May 2008, pp. 847-850.
- [6] Y. Shi and W. C. Karl, “Real-time tracking using level sets”, in *Proc. of IEEE Conference on Computer Vision and Pattern Recognition Workshop*, vol. 2, Jun. 2005, pp. 34–41.
- [7] E. Meijering, I. Smal, and G. Danuser, “Tracking in molecular bioimaging”, *IEEE Signal Processing Magazine*, vol. 23, no.3, May 2006, pp. 46-53.
- [8] K. Li and T. Kanade, “Cell population tracking and lineage construction using multiple-model dynamics filters and spatiotemporal optimization”, in *Proc. of the 2nd International Workshop on Microscopic Image Analysis with Applications in Biology (MIAAB)*, Sep. 2007.
- [9] D. House, M.L. Walker, Z. Wu, J.Y. Wong, and M. Betke, “Tracking of Cell populations to understand their spatio-temporal behavior in response to physical stimuli”, in *IEEE Conference on Computer Vision and Pattern Recognition Workshop on Mathematical Modeling in Biomedical Image Analysis (MMBIA 2009)*, Jun. 2009, pp. 186-193.
- [10] M. Primet, A. Demarez, T. Francois, A. Lindner, and L. Moisan, “Tracking of cells in sequence of images using a low-dimension image representation”, in *Proc. of IEEE International Symposium on Biomedical Imaging (ISBI)*, May 2008, pp. 995-998.

- [11] D. J. Stephens and V. J. Allan, “Light microscopy techniques for live cell imaging”, *Science*, vol. 300, no. 5616, Apr. 2003, pp. 82-86.
- [12] K. Li, E. Miller, L. Weiss, P. Campbell, and T. Kanade, “Online tracking of migrating and migrating and proliferating cells imaged with phase contrast microscopy”, in *Proc. of CVPRW’06*, 2006, p.65.
- [13] K. Althoff, J. Degerman, and T. Gustavsson, “Combined segmentation and tracking of neural stem-cells”, in *Scandinavian Conference on Image Analysis*, 2005, pp. 282-291.
- [14] M. Sonka, V. Hlavac, and R. Boyle, “*Image Processing, Analysis, and Machine Vision (Third Edition)*”, pp. 123-131, 144-146, 212-216, 180-181.
- [15] G. Bradski and A. Kaehler, “*Learning OpenCV: Computer Vision with OpenCV Library*”, pp. 109-113, 234-239, 123-124, 153-157.
- [16] D. Bradley and G. Roth, “Adaptive thresholding using the integral image”, *Journal of Graphics Tools*, vol. 12, issue Feb. 2007, pp. 13-21.
- [17] R.C. Gonzalez and R. E. Woods, “*Digital Image Processing (3rd Edition)*”, pp. 672-674, 733-738.
- [18] R. O. Duda and P. E. Hart, “Use of Hough Transformation to Detect Lines and Curves in Pictures”, *Commun. ACM*, vol. 15, no. 1, Jan. 1972, pp. 11-15.
- [19] J. Candamo, R. Kasturi, D. Goldgof, and S. Sarkar, “Detection of Thin Lines using Low-Quality Video from Low-Altitude Aircraft in Urban Settings”, *IEEE Trans. Aerospace and Electronic Systems*, vol. 45, no 3, Jul. 2009, pp. 937–949.
- [20] J. Iisaka, J. H.H. Bloedow, and T. Sakurai -Amano “Automatic Detection of V-Shaped Features from SAR Imagery”, in *Proc. of ACRS 1996*.
- [21] F. Ren, J. Huang, Mutsuhiro Terauchi, Ruyi Jiang, and Reinhard Klette, “Lane Detection on the iPhone”, *Arts and Technology, Copyright 2010*, vol. 30, pp. 198-205.
- [22] <http://opencv.willowgarage.com/wiki/>. Accessed 1/31/2010.
- [23] M.T. Shaikh, C. Teng, J. K. Archibald, L. R. Sperry, and M. D. H. Hansen, “Automatic Identification and Tracking of Retraction Fibers in Time-Lapse Microscopy”, in *Proc. of 22nd IEEE International Symposium on Computer-Based Medical Systems, CBMS 2009*, Aug. 2009, pp. 1–4.