Segmentation of cell structures in fluorescence confocal microscopy images

by

Hong Gao

A thesis submitted in partial fulfilment for the requirements for the degree of Master of Philosophy at the University of Central Lancashire

March/2013

Student Declaration

Concurrent registration for two or more academic awards

Either *I declare that while registered as a candidate for the research degree, I have not been a registered candidate or enrolled student for another award of the University or other academic or professional institution

or ~~*I declare that while registered for the research degree, I was with the University's specific permission, a *registered candidate/*enrolled student for the following award:~~

_____

Material submitted for another award

Either *****I declare that no material contained in the thesis has been used in any other submission for an academic award and is solely my own work

or ~~*I declare that the following material contained in the thesis formed part of a submission for the award of~~

_____

(state award and awarding body and list the material below):

** delete as appropriate*

Collaboration

Where a candidate's research programme is part of a collaborative project, the thesis must indicate in addition clearly the candidate's individual contribution and the extent of the collaboration. Please state below:

Signature of Candidate .....Hong Gao.....

Type of Award **.....**MPhil of Engineering**.....**

School **.....**School of Computing, Engineering and Physical Science**.....**

Abstract

During the past several years, image segmentation techniques have been developed and extensively used in biomedical applications as an important tool to extract objects and boundaries of interest. In biological field, cytoskeleton analysis is a complicated problem and the analysing technique is still immature. Cytoskeleton plays an important role in normal cell activities, including motion and division, which make the cell cytoskeleton important to investigate. The objective of this project is to investigate and evaluate level set segmentation methods for segmentation of both cell nuclei and membrane segmentation of microfilament images captured by fluorescent confocal microscopy. Based on some background investigations, the active contour methodology has been selected as the fundamental method for image segmentation. This thesis presents the methods used and reports on the results achieved for cell and nuclei segmentation using the hybrid level-set method and cell membrane segmentation using the subjective surfaces model. In addition, some initial results of nuclei segmentation in 3-D case based on the hybrid method will be presented as well. Also included in this thesis are the method and the initial categorisation of microtubule images based on the multi-template method. At the end of the thesis, possible directions for potential future work are presented. It is envisaged that the segmentation tools produced by the project will make cell cytoskeleton data analysis much more convenient. In particular, the

segmentation of cell membranes will help biologists to perform quantitative analysis of fluorescent confocal microscopy images by measuring the cell properties. With more useful information of cytoskeleton being provided, the work contained in this thesis has the potential to contribute to evaluation and prediction of the possibility of cell canceration.

Contents

List of figures

List of tables

List of symbols

*	convolution
$\overline{y}$	rod kernel intensity average
AOS	additive operator splitting
C	initial contour
C_0	object boundary
C_{ij}	correlation coefficient
CM	Confocal microscopy
cot	cotangent angle
D	polarity
E	energy function
F_1, F_2	fitting terms
FCM	Fluorescent confocal microscopy
g	edge indicator
G_σ	Gaussian kernel with standard deviation
H	mean curvature
I_{ij}	mean image intensity
IMF	Intermediate filament
inf	infinite

MF	Microfilament
MT	Microtubule
NSD	normalised standard deviation
PMT	Photomultiplier tube
S	hypersurface
SD	standard deviation
tan	tangent angle
x_i	neighbouring image intensity
y_i	rod kernel intensity
δ_i, δ_j	2-D pixel displacement
θ_k	rod kernel rotation orientation
θ_M	mean angle
τ	time step
Φ	segment contour function
Φ_0	initial contour function
ϕ	segment contour

Acknowledgements

Firstly, I would like to thank Dr Bogdan Matuszewski, my Director of Studies, for his consistent support and guidance through all stages of this research project, to which he has generously devoted much time and effort.

I would like to thank my second supervisors Professor Lik-kwan Shark who generously devoted time to reading the manuscript, and who made many valuable suggestions for its improvements.

I would like to thank Dr Mark F. Murphy, Professor David Burton, and all the researchers of the General Engineering Research Institute from the Liverpool John Moores University, who had provided the data for me and generously devoted time to explaining them and discussing with me. This project was supported by the Engineering and Physical Science Research Council [grant number EP/H024913/1].

I would like to thank my parents, families and all my friends, for their faithful support, advice, and love during my time of studying.

Statements

The program used for microtubule segmentation (in chapter three) is implemented by me. The programs used for microfilament segmentation (in chapter four) are originally implemented by Dr Yan Zhang, modified to deal with microfilament data by me, and owned by UCLan Applied Digital Signal and Image Processing Research Centre (CEPS).

I confirm that I have obtained permissions from the authors by emails to use all the figures obtained from reference sources, and the numbers of references are noted in the figure commands.

Chapter one

Project introduction

1.1 Project background

Cytoskeleton is cellular scaffolding existing in all cells, and it is a dynamic three-dimensional structure that fills cytoplasm [1]. The segmentation of cellular structures is an essential task in cell imaging as it provides information about cell morphology [2]. It enables measurements which can be used to analyse cell differentiation, to create a lineage tree and to determine a cell proliferation rate, to name a few. This research has been carried out in a context of cell cytoskeleton organization analysis. . By providing the image segmentation tools to assist the biologists to measure and analyse cell properties, the results of this research could facilitate a large scale investigation of effects that ionising radiation insult has on cytoskeleton structure, thereby offering a better understanding of cells bio-mechanical responses during cancer radiation therapy [2].

The basic structure of cytoskeleton images and the instrument used for image acquisition will be briefly introduced in this section. As a part of the “Technology in Radio Therapy Feasibility Studies” project, the images used in this project have

been provided by the General Engineering Research Institute from the Liverpool John Moores University.

There are two different type of images used in this project, microtubule and microfilament images. However, the main focus of this research is the microfilament images due to its important role in cell analysis. Figure 1.1 shows a microtubules image on the left a microfilament image on the right. From the figure, it can be found that a microtubule image has more distinct and complete fibre constitutions of cells than a microfilament image. A microfilament image shows more of cell shapes and the membrane parts are clearer than those in a microtubule image.

Microtubule data Microfilament data

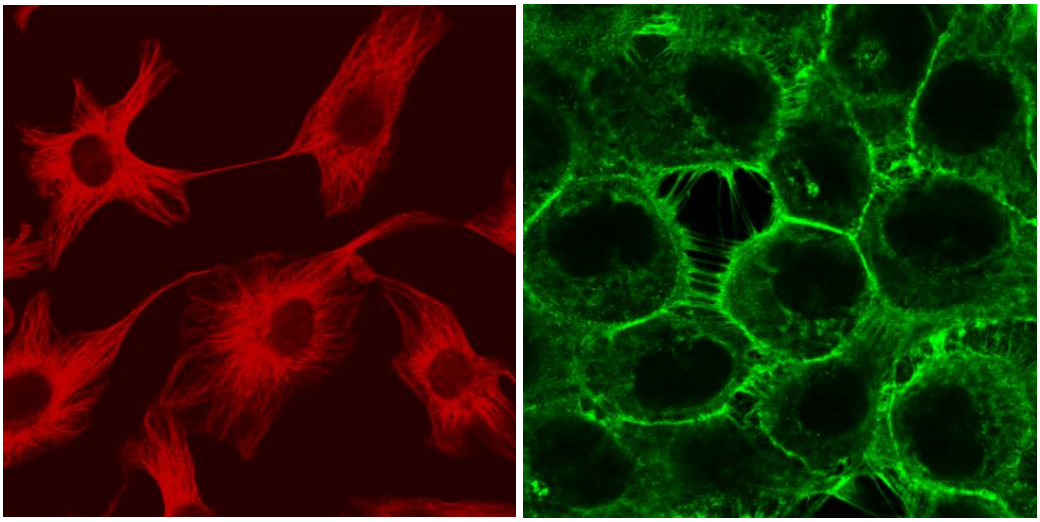


Figure 1.1 Two different types of cytoskeleton data

Microtubule data are composed of numbers of complex fibre constitutions. Each fibre constitution is individually along one fixed direction, and the direction will not be changed. The challenge of processing this data type is to segment every individual fibre constitution accurately. The method introduced in chapter three

managed to conquer this obstacle and segment the microtubule structure successfully. Microfilament data are composed of complex protein constitutions, and these constitutions do not have established directions. This property of microfilament proteins makes this data type reveals more about the shape or form of a whole cell. The challenge of processing microfilament data is to segment cells with relatively indistinguishable and missing boundaries. The methods explained in chapter four overcome the difficulty and segment both cell nuclei and membranes successfully.

1.1.1 Introduction to image segmentation

Segmentation is an important image processing technique and the main purpose is to separate target objects or regions from backgrounds. This can be achieved by clustering the input digital image into multiple salient areas [3]. Since segmentation can make an image easier to be processed further by higher level processing techniques, it is often used as pre-processing in many image analysis procedures [3].

Nowadays, image segmentation technique has been widely used in object detection and recognition, image editing, image compression, and image database search [4]. For example, segmentation is used as a pre-processing section of object recognition, i.e. face, iris, finger-print recognition etc., to locate or detect the target objects [3]. In addition, in traffic, meteorological, military and medical area, image segmentation is also becoming a vital technique [5].

For traffic image analysis, segmentation is normally used to locate and separate the target cars in the acquired images, so that some further work, such as car plate number recognition, can be continued [6]. In remote sensing area, segmentation technique plays a key part in satellite image processing, like city landform analysis and crop disease prediction. In the meteorological area, clouds location and analysis, and weather forecast are also need the help from segmentation technique [4]. In military field, it needs to segment targets to provide parameters for the automatic recognition systems. It can also help provide evidence for precise navigation and guidance of aircrafts [4].

In medical science fields, image segmentation techniques are widely used to segment organs, for instance, brain, heart, lungs etc., and also cells [5]. In particular, image segmentation can assist disease diagnosis by dividing different organs into separate regions. By extending the image segmentation techniques to three dimensions, it can be used for organ reconstruction [7]. With the project focusing on segmentation of microfilament images, further data detail of the image will be described in chapter two and the algorithm utilised in chapters three and four.

1.1.2 Introduction to cytoskeleton

Cytoskeleton is usually defined as a cell's complex internal framework or scaffolding of protein filaments [8]. All the cell-shape determination, cell movement, cell division, and internal transport involve cytoskeleton. Cytoskeleton does not only play an indispensable role in both cell structure and

normal cellular operational function, but also provide important information for cell data analysis and possible morphology prediction [9]. It is primarily composed of three different major components: actin filament (which is also known as microfilament), microtubule, and intermediate filament [10]. The first two classes of filaments, i.e. microfilament and microtubule, are participated in intracellular transport. On the contrary, the third polymer provides mechanical toughness to structures instead of being involved in transport [8].

Microfilaments are plentiful protein in most eukaryotic cells, and have a mechanically supportive function in determining cells' surface shape [11]. In addition, microfilaments participate in the cell motion [11]. Therefore, microfilaments contribute to cell function and structure, studying microfilaments will help to understand cells better, which makes microfilaments the focus part of this project. Microtubules are made by polymerization of tubulin protein subunits, and determine the position of membrane-bound organelles [12]. Moreover, microtubules also direct the intracellular transport process [13]. Intermediate filaments provide mechanical strength and resistance to shear stress [13]. They are made of intermediate filament proteins, which compose a large and heterogeneous family [12].

1.1.3 Introduction to fluorescent confocal microscopy

Fluorescent Confocal Microscopy (FCM) has been used to acquire the cytoskeleton image. Confocal Microscopy (CM) is a special optical imaging technique of light microscopy which started being used in biomedical field since

last century [15]. Using point illumination and a spatial pinhole to eliminate out-of-focus light in specimens (thicker than the focal plan), CM can be utilised to increase optical resolution and contrast of a micrograph. It enables the reconstruction of three-dimensional structures from the obtained images [16]. Nowadays, fluorescence illumination has become the most promptly expanding microscopy technique in the field of biomedical science [14]. "*Fluorescence confocal microscopy (FCM) is a version of confocal microscopy in which the inspected specimen is doped with a high-quantum-yield fluorescent dye that strongly absorbs at the wavelength of the exciting laser beam* [15]." It is now possible to visualize features in living cells and tissues by the aid of FCM technique. In addition, it has also been successfully applied in flow cytometry and even for single molecule detection [14].

Figure 1.2 Internal structure of a FCM (After [17])

Figure 1.2 [17] shown above illustrates the structure of a basic fluorescent confocal microscopy. In such technique, the inspected specimen is marked with a fluorescent dye bonding to the targeted structures. The dye strongly absorbs the wavelength of the exciting laser beam. Subsequently the excited dye molecules fluoresce at a longer wavelength reveals indirectly structures of interest. The fluorescent confocal microscopy makes it possible to visualize features in living cells and tissues. Figure 1.2 shown above illustrates the structure of a basic fluorescent confocal microscopy. The 3D volume data acquired by fluorescent confocal microscopy compose of multiple 2D image slices, as illustrated in figure 1.3. As an example, figure 1.4a shows an acquired image stack of monolayer from microtubule data and figure 1.4b shows different slices of them.

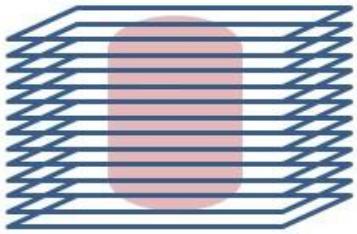


Figure 1.3 3D volume data composed of multiple slices

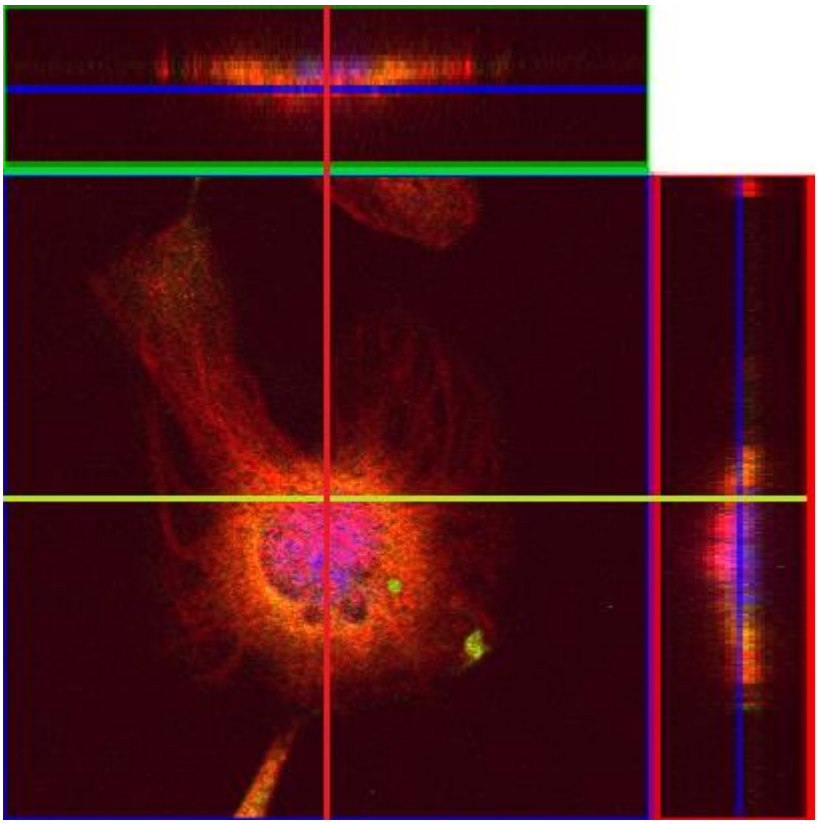


Figure 1.4a Image stack showing monolayer of 3D microtubule data

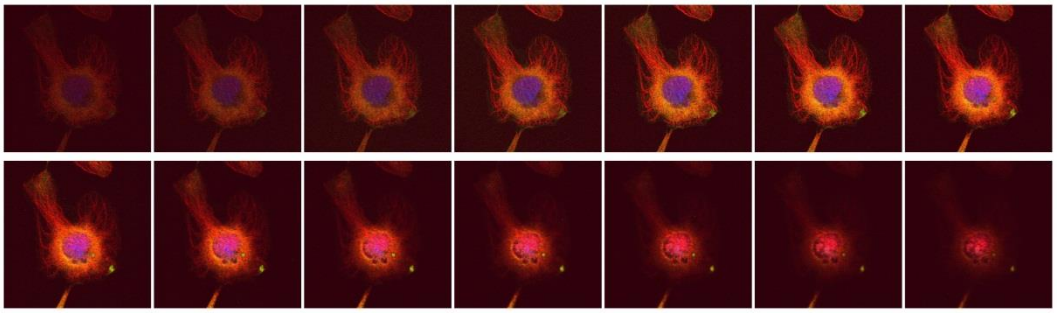


Figure 1.4b Different slices from 3D microtubule data

1.2 Project aim and objectives

This project aims to study the segmentation methods applied to microtubule and microfilament images. The specific objectives of this project are:

a) Understand and segment microtubule image.

b) Segmentation of cell nuclei using level set method.

c) Membrane segmentation using level set method.

In order to review the segmentation procedure, several methods are introduced. To understand the method further, the most important formula is derived. In addition, some algorithms will be introduced to improve the accuracy of the segmentation results.

In this project, segmentation of microtubule data is realised by using multiple templates method. This process categorises microtubule organization and marks orientation on each fibre constitution of the data. Segmentation of microfilament data includes two parts: cell nuclei segmentation and cell membrane segmentation. Cell nuclei segmentation is realised based on active contours without edges model, and cell membrane segmentation is achieved based on subjective surfaces model. This project will focus on investigating the foregoing algorithm and evaluating the segmentation results.

The potential users of this project may include biologists who need to have experiment on cytoskeletons. In order to investigate the variation and predict the cancellation possibility of the cells, it is important to implement a reliable method for cells analysis. If the cytoskeleton segmentation accuracy is high, the cell structure analysis and cell movement observation would be processed more easily. Cell nuclei and membrane segmentation, which play as pre-processing section before the precise and professional cell analysis, make this research meaningful and helpful. There are also some researchers and students who investigating segmentation method for familiar data might have interest in this

research study. This research could help them know more details about basic microfilament segmentation based on level set method.

1.3 Thesis organisation

This project thesis will firstly give briefly introduction of background knowledge, such as image segmentation, cytoskeleton data, etc., and then focus on investigating specific methods and evaluate the segmentation results. The thesis is composed of five chapters stating as followed.

Chapter one gave an introduction of the project and background knowledge related to. In chapter two, some literature review on image segmentation with introduction on several methods will be presented. For the following chapter three and chapter four will focus on specific methods for corresponding data and showing the results. At the end, some future work and conclusion will be given in chapter five.

Chapter two

Literature review on image segmentation

Segmentation is an important technique in image processing and analysis area, which is also a classical and continued problem in computer vision research. So far, there is no such a method that is suitable for every kind of images or a kind of image can be correctly segmented by all methods [5]. Over the years, image segmentation techniques have been improving by cooperating with other new theories and methods in order to deal with different varieties of data types [4]. In the following, some most popular segmentation methods will be briefly introduced.

2.1 Image intensity thresholding methods

Thresholding is the simplest and most often used method compared with all the other segmentation methods. It changes a grey image into a binary image based on a threshold value which can be automatically or manually selected. Since the segmentation results are controlled by the threshold value selected, choosing appropriate threshold values is pivotal for this method [18]. If a threshold value is set as T, then the output binary image will be defined as $F(x,y) = \begin{cases} 1 \; if \; I(x,y) > T \\ 0 \; if \; I(x,y) < T \end{cases}$, (2.1) where I(x, y) represents the original input image. In the

following figure 2.1, shows an example of the segmentation result produced by thresholding.

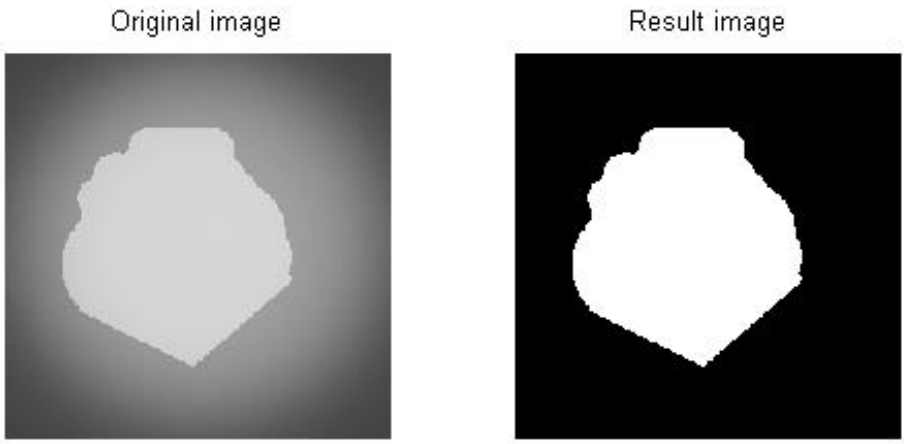


Figure 2.1 A basic thresholding segmentation example

For the input grey image shown on the left of figure 2.1, the output binary image, showing on the right figure, is generated by setting a threshold value to T = 180. The threshold value is defined by using the intensity histogram of the original image. As shown in figure 2.2 below, the histogram of the original image consists of two groups of intensity values corresponding to bright object and dark background, and the threshold value selected for segmentation falls in the gap (circled in red) of the two intensity groups.

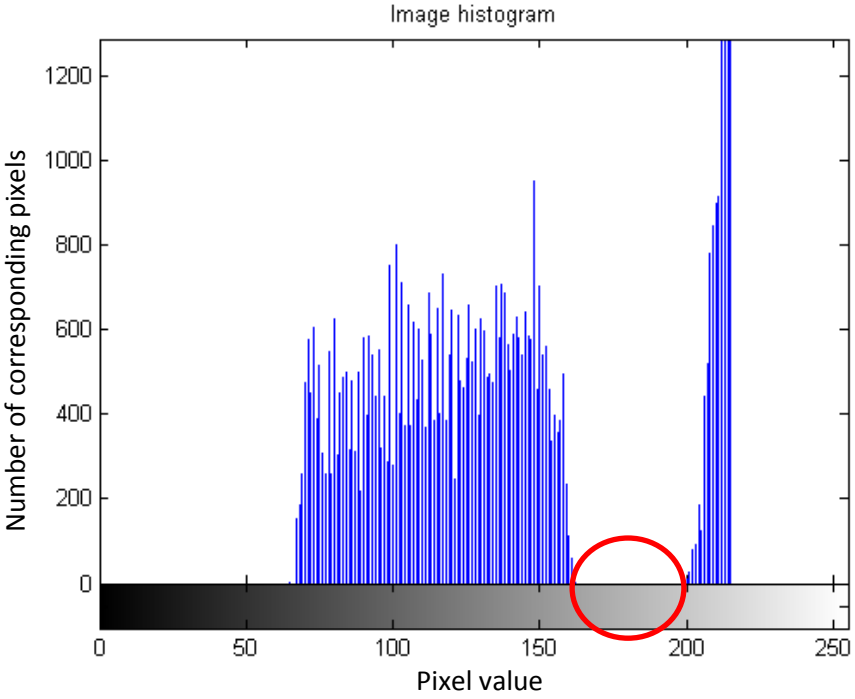


Figure 2.2 Original image intensity histogram

Figure 2.3 shows a simple example of applying thresholding method to microfilament data. It can be found that this simple thresholding method presents a poor segmentation result. Modifying an adaptive threshold [19] value with aid of noise elimination may gain a better result. This method is just base on the image intensities, which is not suitable for the provided data.

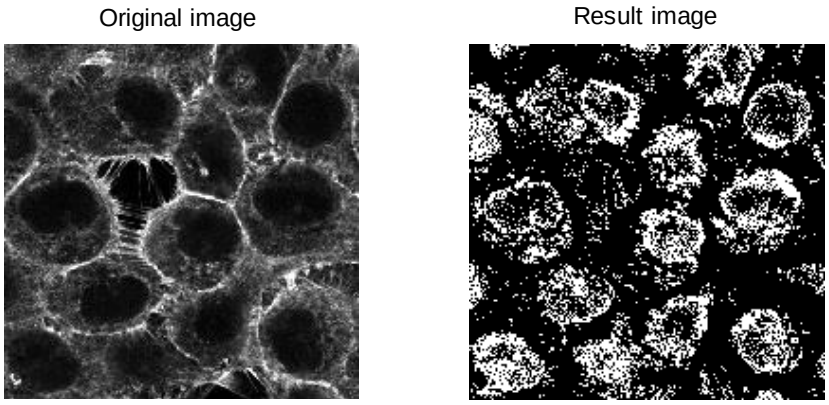


Figure 2.3 Example of cell segmentation using image intensity thresholding method

2.2 Region growing methods

The region growing method [20] was first considered as the seeded region growing method, which takes a group of seeds as input along with the image to be segmented. Each of the target objects to be segmented is marked by a corresponding seed. During the procedure, the regions will be iteratively grown by comparing all unallocated neighbouring pixels with the pixels inside the regions. The difference, which is denoted as δ, between the intensity value of the neighbouring pixels and the mean intensity value of pixels inside the regions, will be used to measure similarity. The pixel with a sufficiently small difference value will be allocated to the respective region and the procedure will complete after all pixels are allocated to its corresponding region. A seeded region growing method [6] [20] requires seed pixels as the additional input, which leads the segmentation results to be dependent on the chosen seed. Therefore, seeded region growing is influenced by noise so that some regions of interest may not be correctly segmented [5].

A modified algorithm, called unseeded region growing [20], realises the region growing without the requirement of explicit seeds. This method starts off with a single region Ω_1, and the seed pixel chosen here will virtually not affect the final segmentation result. During each iteration time, this unseeded region growing method will compare the neighbouring pixels in the same way as that being processed by seeded region growing [21]. The difference between these two region growing methods is in setting the threshold value. For the unseeded region growing method, if the minimum value is less than a predefined threshold

value, then the corresponding pixel can be allocated to the respective region. Otherwise the pixel will be considered different from all current regions and a new region will be created based on this pixel [20]. But for seeded region growing methods, the number of the regions is equal to the number of input seed pixels and it will not change.

Figure 2.4 [22] shows an example of seeded region growing segmentation. The left figure present the original cell image and the right figure shows the segmentation result. In the original image, background is marked in pearl blue, cells are marked in blue, and those parts marked in royalblue represent cell nuclei. In the result image, background, segmented cells and nuclei are marked in white, grey, and black, respectively. The image used in this example contains three cells. This method produces a good segmentation of cells with aid of K-means clustering as pre-processing. It also needs gradient information of input image. Consequently, this method can only segment well for cells with distinct boundaries.

Figure 2.4 Example of cell segmentation using region growing method (source: [22])

2.3 Clustering methods

These methods consider segmentation as a clustering issue and have been commonly used in image area segmentation. Some familiar algorithms, such as k-means algorithm, fuzzy c-means algorithm, ISODATA algorithm, are widely used in clustering based segmentation [5]. As an example, the k-means algorithm is presented in this section.

Normally, complex interactions with existing clusters are used in common clustering methods to give accurate results [6]. The k-means algorithm is a reduplicative process that will separate the input image into k clusters and the basic processing procedures are explained as follows.

In the first step, the k cluster centres are chosen randomly or based on some initial analysis. In second step, every pixel of the input image is assigned to its nearest cluster based on the distance to cluster centres. In the third step, the new cluster centres are estimated by obtaining the averages for all the clusters from all their pixels. The two steps, distance minimisation and cluster centre re-computation, are repeated until no more pixels change to other clusters [6]. The quality of segmentation results depends on the number of clusters and the initial set of clusters. This k-means algorithm is guaranteed to converge but the result may not be as accurate as expected. In figure 2.5 [23], the segmentation is obtained by using k-means method to find the optimal threshold value. Subfigure on the left shows the original input image, and the segmented result is shown in the right subfigure. In the original image, the small grey dots represent

cells and background is marked in black. In the result image, the background and segmented cells are marked in black and white, respectively. This method is performed using the histogram of image intensity, but the provided images contain missing boundaries, which means this method cannot be suitable to segment the cytoskeleton data.

Figure 2.5 Example of cell segmentation using clustering method (source: [23])

2.4 Active contours

Active contours are widely used for segmentation in varieties of areas. Active contours lock onto nearby edges and localise them accurately. This section introduces two different models of active contours: parameter active contours and level set.

2.4.1 Parametric active contours

Parametric active contours are also called snakes, which was proposed by M. Kass [24] in 1987. A snake is an energy-minimizing spline guided by external

constraint forces and influenced by image forces that pull it toward features such as lines and edges [24]. Snakes provide a unified account of a number of visual problems, including detection of edges, lines, and subjective contours; and have been applied to motion tracking; and stereo matching [24]. The snake model is a dynamic procedure as it always minimising the energy function. The most significant difference between snakes and most other methods is that the model is active rather than finding salient contours.

The basic snake model is a controlled continuity spline under the influence of image forces and external constraint forces [24]. The internal spline forces serve to impose a piecewise smoothness constraint. The image forces push the snake toward salient image features. The external constraint forces are responsible for putting the snake near the desired local minimum. Representing the position of a snake parametrically by $v(s) = (x(s), y(s))$, the energy function can be expressed as:

$$E^*_{snake} = \int_0^1 E_{snake}(v(s))ds = \int_0^1 E_{int}\big(v(s)\big) + E_{image}\big(v(s)\big) + E_{con}(v(s))ds \quad (2.2)$$

where E_{int} represents the internal energy of the spline due to bending, E_{image} gives rise to the image forces, and E_{con} gives rise to the external constraint forces.

Although snake model can provides more accurate segmentation results than most other method, some shortcomings exist. It requires the initial contour to be set manually by the user, and the contour needs to be set at the outside of the

target object. In this case, the segmentation result depends on the position of the initial contour. Moreover, multiple objects require to be segmented respectively rather than being segmented at the same time. Therefore, a snake model is not suitable for segmenting microfilament images as they contain multiple cells.

Figure 2.6 [25] shows an example of cell segmentation using snake model. The left figure shows the original data and the right figure shows the segmentation result. As illustrated in figure 2.6, it can be observed that the segmentation is not accurate. Manual selection of initial contour is required, which will affect the segmentation result. In the result image, the segmented cell is marked in blue, and background is in yellow. Therefore, another active contours model, which improves the model and present better results, will be introduced next section.

Figure 2.6 Example of cell segmentation using snake model (After [25])

2.4.2 Level-set

The level-set method was firstly put forward by Osher [26] in 1988 for moving interfaces tracking, and had spread to its wide use in various image processing procedures in the past decades. The main idea of the level-set method is energy minimisation. It represents an evolving contour with the aid of the predicted function, with zero level being the actual contour. Furthermore, a similar flow for the implicit surface can be derived according to the contour equation. It will reflect the propagation of the evolving contour when the derived flow is applied to the zero level [26].

Rather than following the interface, the level set method builds the contour into a surface instead of taking the original curve [27]. In figure 2.7 [28], where the original curve is shown as the red circle on the left and the level-set case on the right, the green cone-shaped surface shown on the right intersects the xy plane exactly where the contour is located [28]. The method accepts any point as input and returns its height as output, and the red front contains all the points with the same height at zero [28]. Therefore, this surface is named the level-set function and the red circle is named the zero level-set [28].

Figure 2.7 Level set function (After [28])

Level-set is an implicit method, which means that it represents the interface curve by a function. The initial contour of this method is free to be set in any position and it can automatically change the topology [29]. Compared to parametric active contour, it allows segmentation for multiple objects at the same time [29]. Chan-Vese model is a widely used level-set segmentation method in many research fields. This model is considered to perform well for microfilament segmentation and will be explained further in chapter four.

In figure 2.8 [30], some cell segmentation results produced by a level-set method are presented. The figure shows two different segmentation results of using level-set method, and the segmented cells are circled with white contours. Although the images contain some indistinct cells, the level-set method manages to segment them accurately. Therefore, level-set method is considered as the preferred method that can be applied to the cytoskeleton images.

Figure 2.8 Example of cell segmentation using level set method (source: [30])

2.5 Graph partitioning methods

Graph partitioning is an important problem that has been extensively applied in many areas [31]. The problem is to partition the vertices of a graph into p roughly equal parts, where the value of p can be set manually by the users. In that case, the number of edges connecting vertices in different parts is minimised. The graph partitioning problem is NP-complete, and large numbers of algorithms have been developed to find a reasonable partition.

Spectral partitioning methods are widely used as they are known to produce good partitions for a wide class of problems [32] [33]. However, these methods are very expensive since they require large complex mathematical computation. Another graph partitioning technique called geometric partitioning algorithms [34] [35] tend to be fast but the partition results are worse than those obtained by spectral methods.

Another class of graph partitioning algorithms that can provide better partition results than spectral methods with a lower computation cost is called multilevel graph partitioning schemes [36] [37]. In a multi-level graph partitioning algorithm, each stage reduces the size of the graph to partitions of the smaller graph which are then mapped back to refine the partition of the original graph [38]. A wide variety of partitioning and refinement methods can be applied to the overall multi-level scheme. In most cases, this approach can produce segmentation results with high quality in fast execution times.

Figure 2.9 [39] compares a graph partitioning method and an active contour method for cell image segmentation. For the original image shown on the left, the segmentation results produced by the graph partitioning method and the active contour method are shown in the middle and on the right, respectively. The cells are marked in grey and background is marked in black in the original image. Yellow contours and red contours are correspondingly marked for segmenting cell regions from the background in the result images. It can be seen that the cell contours produced by the active contour method are smoother.

Figure 2.9 Example of cell segmentation using graph partitioning and active contours (source: [39])

2.6 Concluding remarks

Among the image segmentation methods presented in this chapter, image intensity thresholding methods based on a fixed threshold value is the simplest one. The threshold value can be set as a percentage of maximum pixel value found in the input image. As this method is based on the image intensities, it will not be considered as the preferred method. Region growing methods are commonly used for cell segmentation and tracking. An easy way of applying this method to cell images is to manually mark a seed pixel on the target cell.

However, it requires edges information of input image, which induces segmentation of objects with missing boundaries. Clustering methods are normally used in pre-processing stages to improve segmentation results. K-means, which is a commonly used method, is performed using the histogram of image intensity. In this case, this method cannot be used to segment cells without complete boundaries. Although both graph partitioning methods and active contours can produce good results for cell segmentation, active contour methods can automatically complete the segmentation procedure with accurate and smooth contours. Furthermore, graph partitioning methods require manual markers to accomplish the segmentation procedure. Based on the advantages of active contours, it is chosen to be the preferred method in this project.

Chapter three

Segmentation of microtubule images

In this chapter, the algorithm based on a multi-template model, which was used for microtubule segmentation, will be introduced and discussed.

3.1 Approach overview

The multi-template model [9] can be applied to achieve a reliable segmentation of cytoskeleton fibres, and is especially suitable for fuzzy labelled fibre with bright backgrounds. This method examines the possibility of each neighbour pixel to compose a corresponding fibre. The input image is correlated with synthetic templates at different orientations in each corresponding region, and the most significant correlation coefficient among all directions is assigned as the central pixel. All the assigned central pixels create the fibre index. In addition, to reduce the computational complexity of two-dimensional correlation, pairs of perpendicular lines are used in the correlation procedure instead of 2-D template masks.

The correlated data with different masks will be compared to obtain the final data. For each corresponding pixel obtained from correlation with different

masks, the output pixel should be set to the highest intensity value pixel. Figure 3.1 shown below helps to visualise the process.

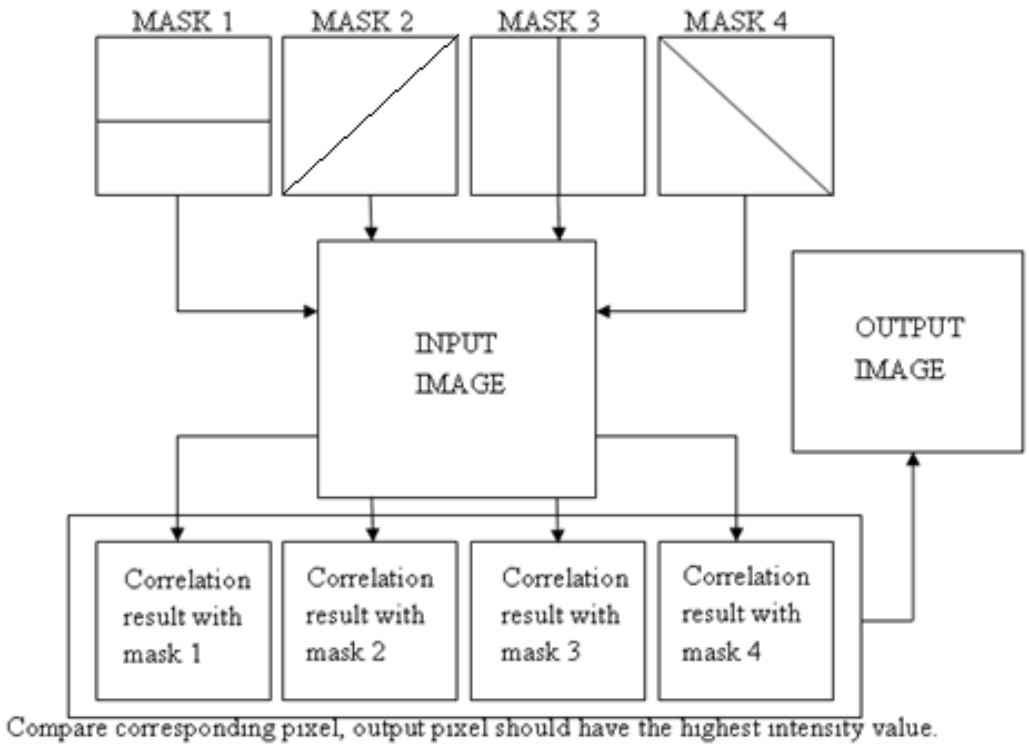


Figure 3.1 The procedure of multi-template correlation

3.2 Background noise elimination

Quantitative fluorescence microscopy is based on the linear relationships between the fluorescent image intensity and the local concentration of the respective labelled epitope [9]. This linearity is often perturbed by sample derived or instrumental contributions grossly referred to as background. Cytoskeletal fibre labelling intensity is often comparable to the local background levels, and thus accurate background elimination as image pre-processing is essential. In this project, two methods are applied for background subtraction: median filter [40] and rolling ball method [10].

Median filter is a common noise filtering method and has been widely used in image processing. As a nonlinear filtering method, it is often used to reduce “salt & pepper” noise and is effective to simultaneously eliminate noise and preserve edges. Figure 3.2 below shows the result of applying a median filter to an image with “salt & pepper” noise.

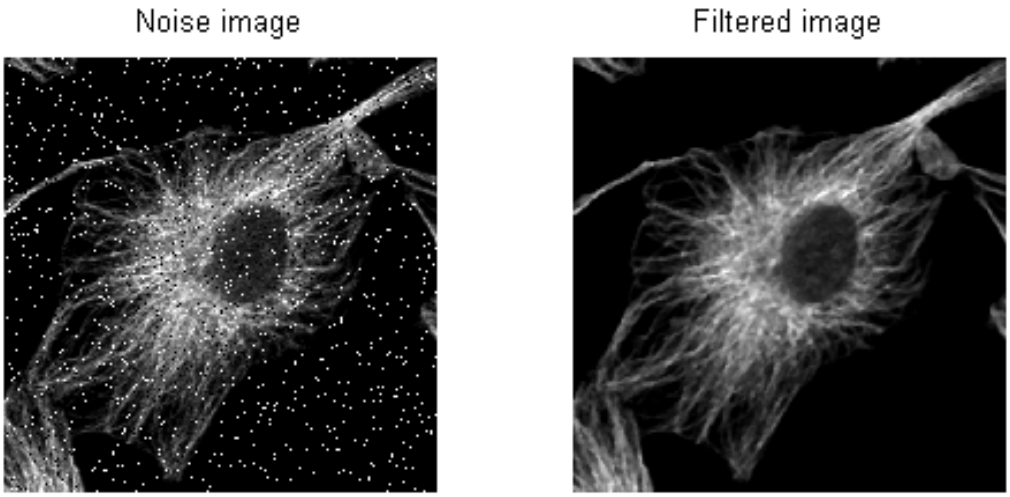


Figure 3.2 “Salt & pepper” noise reduction by median filter

The rolling ball background subtraction method removes smooth continuous background information from input data. In this method, an image is represented by a 3D surface with the height at each pixel position corresponding to its intensity value, and a ball is rolling over the back side of the surface to create the background [10]. By subtracting the background created, the method produces an image with only the information of interest. Figure 3.3 [41] below shows an example with the image background subtracted by the rolling ball method. The left image in figure 3.3 shows the original input, and the image on the right shows the result after rolling ball background subtraction. Subfigure (a)

shows the original input image, and (b) shows the image after background elimination using rolling ball method. Subfigure (c) shows the transverse of the hypothetical 3D surface, and (d) shows the transverse after background elimination.

Figure 3.3 Background subtraction by rolling ball method (After: [41])

For microtubule images, an example is shown in figure 3.4 to illustrate the difference between the images before and after background subtraction. The left image in figure 3.4 shows the microtubule polymer before background subtraction and the right one shows the result after applying the median filter with a 5-by-5 kernel and the rolling ball method with a diameter of 7 pixels.

(a) (b)

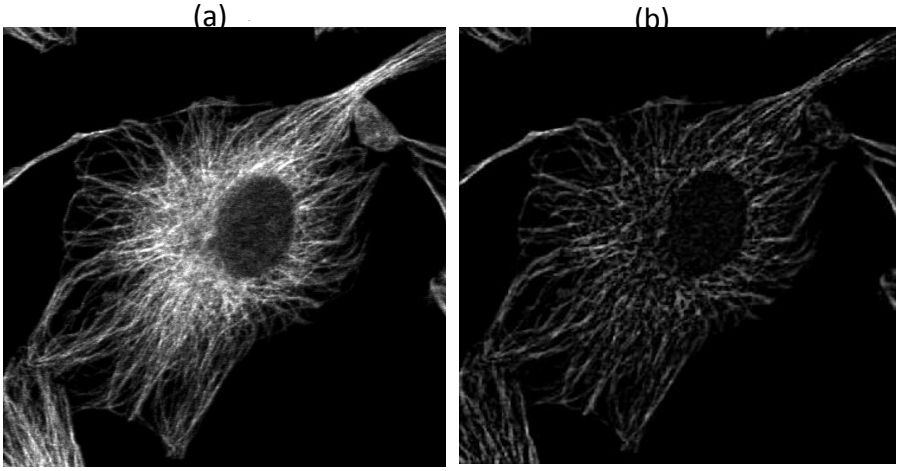


Figure 3.4 (a) Original input image, (b) image after background subtraction

3.3 Multi-template algorithm

Let P(i,j) be the intensity of point (i,j) in a two-dimensional image P, and let each pixel (i,j) be the centre of a 2K+1 length rod kernel at equally spaced orientations θ_k:

$\theta_k = k\Delta\theta_k$, where $\Delta\theta_k = \pi/(2K)$ and k = 0, 1, 2, … , 2K-1.

Let I = -K, … , K, the list of 2K+1 two-dimensional pixel displacements can be expressed as:

$[\delta_i(I, \theta_k), \delta_j(I, \theta_k)] = [I, I|\tan(\theta_k)|]$, if $\theta_k < 45^o$ or $\theta_k > 135^o$;

$$= [I|\cot(\theta_k)|, I], \text{ if } 45^o < \theta_k < 135^o \quad (3.1)$$

This expression represents all the one-dimensional non-zero elements in the (2K+1)×(2K+1) rod kernel for each θ_k.

The intensities of the rod kernel are weighted by a Gaussian profile:

$$y_I = G[\delta_i(I, \theta_k), \delta_j(I, \theta_k)] = e^{\frac{I'^2}{2\omega}}, \quad (3.2)$$

$$I' = I/\max[|\sin(\theta_k)|, |\cos(\theta_k)|] \quad (3.3)$$

$$\bar{y} = \sum_{-K}^{K} \frac{y_I}{2K+1} \quad (3.4)$$

For each pixel (i,j), the neighbouring image intensities along the rod can be expressed as [9]:

$$x_I = P\left[i + \delta_i(I, \theta_k), j + \delta_j(I, \theta_k)\right] \quad (3.5)$$

The mean image intensity value along the rod kernel is calculated as [9]:

$$I_{ij}(\theta_k) = \bar{x} = \sqrt{\sum_{-K}^{K} \frac{x_I}{2K+1}} \tag{3.6}$$

The standard deviation (SD) of the intensities and the normalized standard deviation (NSD) are [9]:

$$SD_{ij}(\theta_k) = \sqrt{\sum_{-K}^{K}(x_I - \bar{x})^2/(2K+1)} \tag{3.7}$$

$$NSD_{ij}(\theta_k) = SD_{ij}(\theta_k)/I_{ij}(\theta_k) \tag{3.8}$$

Therefore, the correlation coefficient at central point (i,j) along the rod kernel is calculated as [9]:

$$C_{ij}(\theta_k) = \sum_{-K}^{K}(x_I - \bar{x}) \times (y_I - \bar{y}) / \sqrt{\sum_{-K}^{K}(x_I - \bar{x})^2 \sum_{-K}^{K}(y - \bar{y})^2} \tag{3.9}$$

Similarly, $I_{ij}(\theta_k + \pi/2)$, $SD_{ij}(\theta_k + \pi/2)$, $NSD_{ij}(\theta_k + \pi/2)$ and $C_{ij}(\theta_k + \pi/2)$ denote the corresponding fibre values for a perpendicular rod.

In order to calculate the quantitative analysis of cytoskeleton fibre, the fibre polarity and length need to be defined. Polarity can be denoted as D and defined as [9]:

$$D = \sqrt{D_x^2 + D_y^2} \tag{3.10}$$

where $D_x = \sum mean(i,j) \cos\left(2\Phi_{ij}\right) / TotI)$ (3.11)

$$D_y = \sum mean(i,j) \sin\left(2\Phi_{ij}\right) / TotI) \tag{3.12}$$

$$TotI = \sum mean(i,j) \tag{3.13}$$

In addition, the mean angle at each corresponding pixel can be defined as [9]:

$$\theta_M = \frac{1}{2acrtan\left(\frac{Dy}{Dx}\right)} \tag{3.14}$$

Total fibre length, F, is defined as [8]:

$$F = \sum I / \max[|\cos(\Phi_{ij})|, |\sin(\Phi_{ij})|] \tag{3.15}$$

where $\Phi_{ij} = orientation(i,j)$, which means the orientation of the highest correlation coefficient rod at point (i,j).

In figure 3.2 shows an example of applying the multi-template method to a simulated test image. The upper left subfigure shows the simulated test image, the upper right subfigure shows the categorisation result, and the bottom of the figure, shows the result with orientation. Based on the colour bar on the right of the bottom subfigure, it can be seen that the lines in the simulated image have been correctly identified with the horizontal line shown in green, 45-degree line shown in blue represents, vertical line shown in purple, and 135-degree line shown in red.

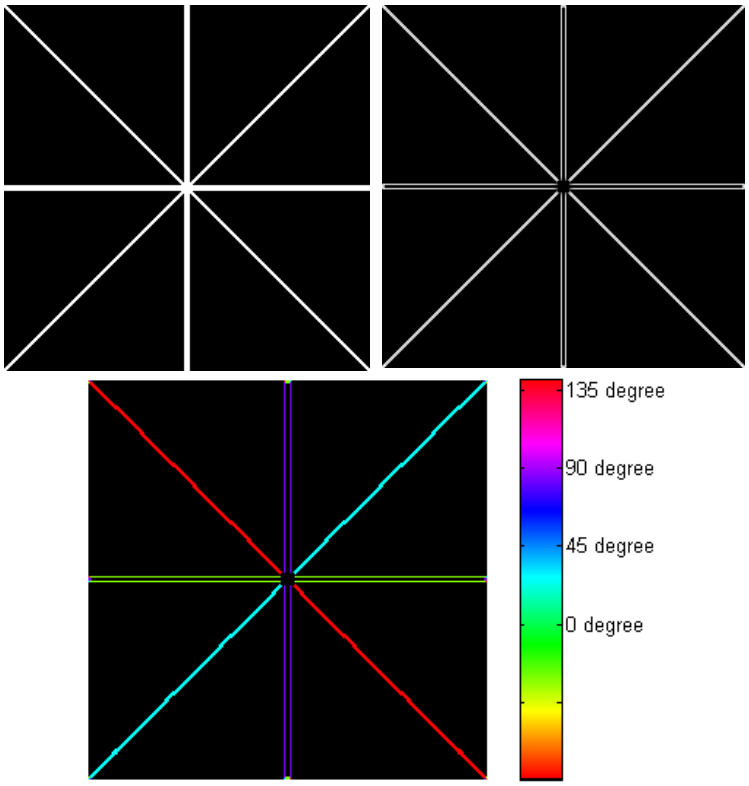


Figure 3.5 Result of applying multi-template model to simulated test image

3.4 Experimental results and discussion

The segmentation results of microtubule images using the multiple template method are shown in figure 3.6. The upper left figure is the original input image after background subtraction, the upper right figure shows the categorization result, and the bottom figure shows the result with the fibre orientation information shown in colour.

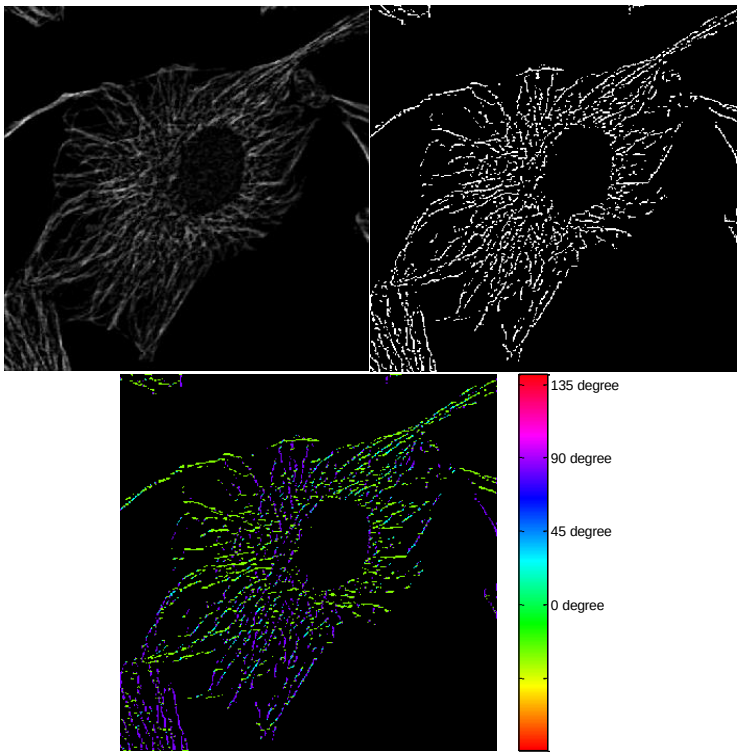


Figure 3.6 Results of applying multi-template model to microtubule data

The results shown in figure 3.6 are obtained by applying a correlation mask with a size of 5-by-5 to the image. Figure 4.2 shows another image obtained by applying the same kernel size.

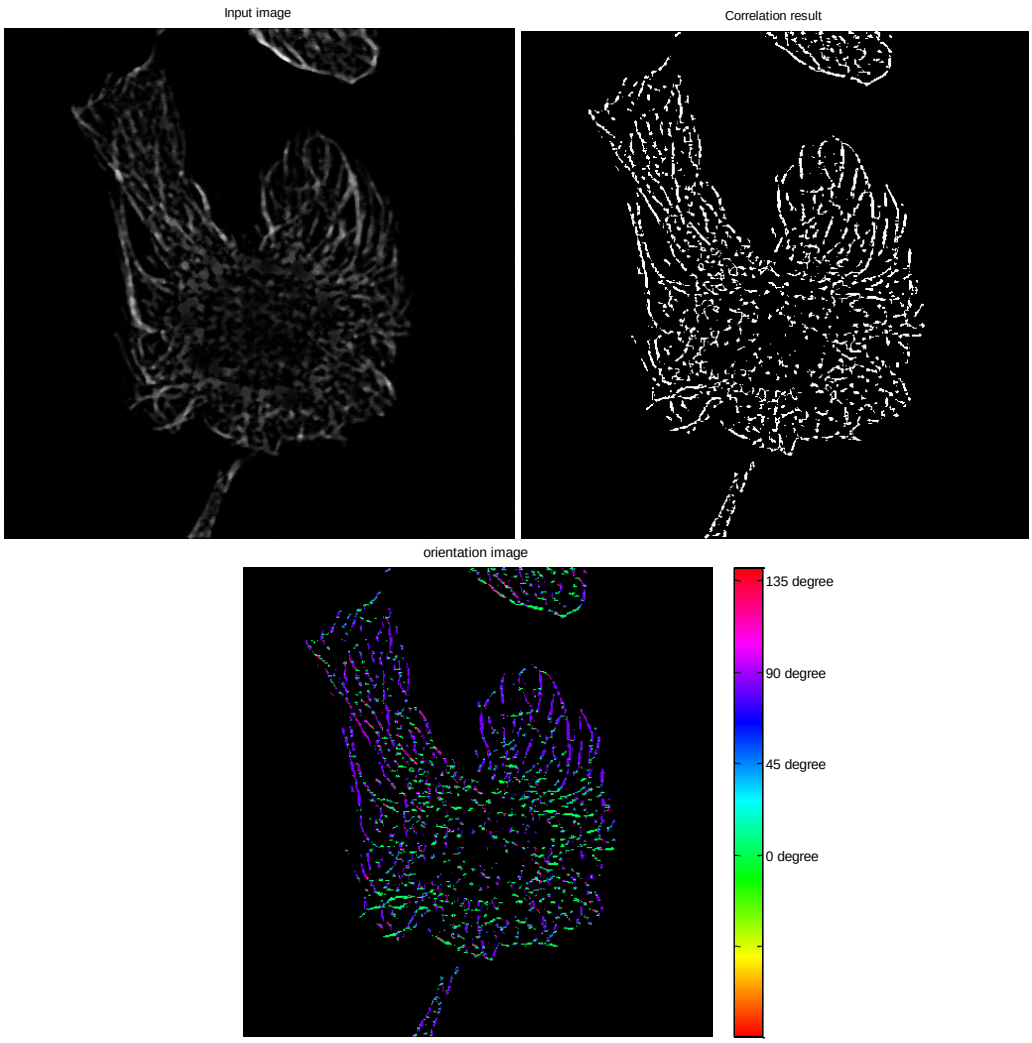


Figure 3.7 Multiple templates method applied to other image

The multiple templates based segmentation of microtubule images realises quantification of the fibre constitutions. This provides a reliable pre-processing method to evaluate microtubule images further. Marking out the orientation presents an aspect of the structure information to the biologists thereby facilitating better analysis of cytoskeleton and better prediction of possible changes of the cells.

3.5 Concluding remarks

Microtubule data normally composes of single fibres with uniform thickness. Morphometric analysis requires fibre intensity per unit length, which can be calculated by dividing the total fibre intensity by the total fibre length [9]. Consequently, fibre intensity per unit length is expected to be a constant [9]. In order to get further understanding of this data type, multiple templates method is used to segment microtubule structure. This test detects the validity of the linear relationship between fibre intensities and the amount of fibres. This analysis applies a powerful tool to study the effects of mutations, drugs, and environmental perturbations on cells. These effects reflect structural and signal transduction responses of the cell, which are presented by their cytoskeletal signature [9]. Although the data provide changed the research focus to microfilament data, this method is a potential direction that can be carried on in future.

Chapter four

Segmentation of microfilament images

4.1 Approach overview

This section focuses on explaining the method used for segmenting microfilament images. For the nuclei segmentation section, a reliable level set segmentation technique called active contours without edges. This method is also known as “Chan-Vese model”. Another segmentation method, named hybrid level set method, will also be explained. This method can also be used for cell nuclei segmentation. Comparing with the Chan-Vese model, this model may have better performance, because it takes object boundaries and additional region information into account. Subjective surfaces model will be used to segment cell membranes, and a detailed description of these methods will be given in this section.

4.1.1 Active contours without edges

The Chan-Vese model is able to detect objects whose boundaries are not necessarily defined by gradient [42]. It is an energy minimisation process which can be seen as a particular case of the minimal partition problem [42].

Let an evolving curve C defined in Ω be the boundary of an open subset ω of Ω, i.e. $\omega \subset \Omega$, and $C = \partial\omega$. In the equation which follows, *inside(C)* denotes the region ω, and *outside(C)* denotes the region $\Omega \setminus \overline{\omega}$. To explain the basic idea of the model in a simple case, it is assumed that image u_0 contains two regions with approximately piecewise-constant intensities of distinct values u_0^i and u_0^o. If u_0^i represents the object region to be detected with C_0 denoting its boundary, then it can be seen that $u_0 \approx u_0^i$ inside boundary C_0, and $u_0 \approx u_0^o$ outside boundary C_0. This gives the following "fitting" term [42]:

$$F_1(C) + F_2(C) = \int_{inside(C)} |u_0(x,y) - c_1|^2 dxdy + \int_{outside(C)} |u_0(x,y) - c_2|^2 dxdy \quad (4.1)$$

where C is any other variable curve, which can be considered as the initial contour in the "fitting" term. c_1 and c_2 are the mean values of u_0 inside and outside curve C, respectively. In this simple case, it should be apparent that the boundary of the object, C_0, is the minimum of the "fitting" term [42]:

$$\inf_C \{F_1(C) + F_2(C)\} \approx 0 \approx F_1(C_0) + F_2(C_0) \quad (4.2)$$

There are four possible cases. If curve C is outside the object, then $F_1(C) > 0$ but $F_2(C) \approx 0$. If curve C is inside the object, then $F_1(C) \approx 0$ but $F_2(C) > 0$. If curve C is both inside and outside the object, then $F_1(C) > 0$ and $F_2(C) > 0$. Finally, the fitting energy will be minimized when curve C is along the object boundary, which means $C = C_0$. These four cases are illustrated in figure 4.1 [42]:

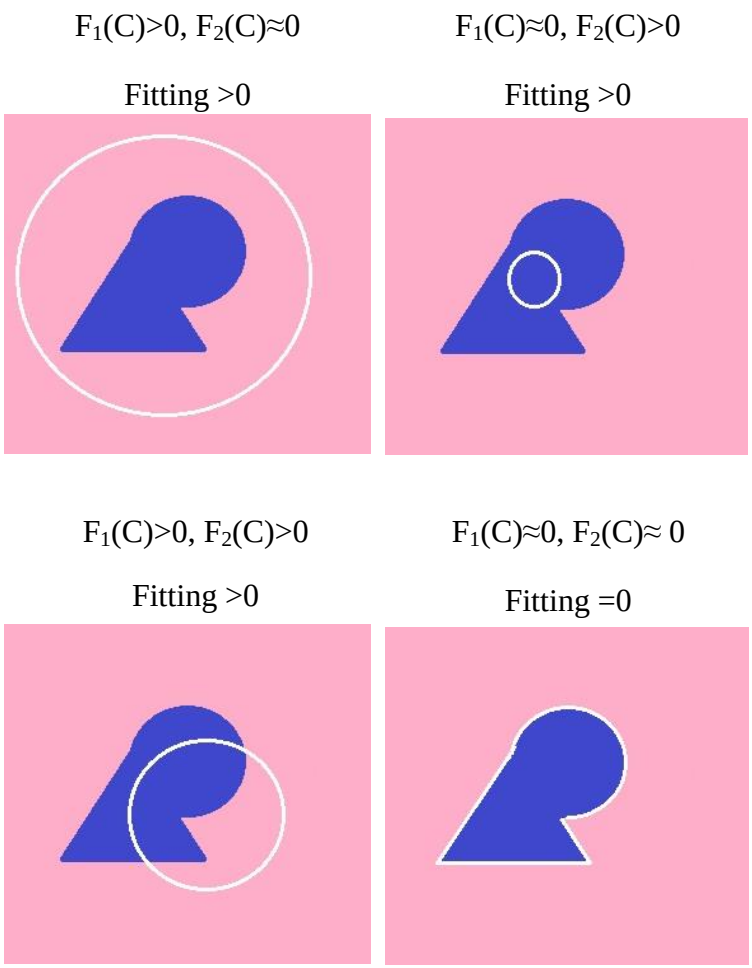


Figure 4.1 All possible cases of the curve positions (modified from [42])

Figure 4.1 shows the relationship between the curve location and the fitting energy. The upper left figure shows the case of the curve located outside the object; the upper right shows the case of the curve located inside the object; the bottom left shows the case of the curve located both inside and outside the object; and the bottom right shows the case of the curve on the object boundary corresponding to the fitting energy being 0.

In the active contour model, the above fitting term will be minimized. In addition, several regularizing terms will be added, such as the length of curve C, and the

area inside the region of curve C. Therefore, the energy functional F(c_1, c_2, C) is given as [42]:

$$\mathrm{F}(c_1, c_2, \mathrm{C}) = \mu \cdot \mathrm{Length}(\mathrm{C}) + \mathrm{v} \cdot \mathrm{Area}\big(\mathrm{inside}(\mathrm{C})\big) + \lambda_1 \int_{inside(C)} u_0(x,y) - c_1|^2 H(\phi(x,y))dxdy + \lambda_2 \int_{outside(C)} u_0(x,y) - c_2|^2 (1 - H(\phi(x,y)))dxdy \quad (4.3)$$

where

$$\mathrm{Length}(\mathrm{C}) = \int_{\Omega} \delta\big(\phi(x,y)\big)\nabla|\phi(x,y)|\, dxdy \quad (4.4)$$

$$\mathrm{Area}(\mathrm{C}) = \int_{\Omega} H\big(\phi(x,y)\big)dxdy \quad (4.5)$$

and where $\mu \geq 0$, $v \geq 0$, $\lambda_1, \lambda_2 \geq 0$ are fixed parameters with manual-set values. Keeping c_1 and c_2 established, and minimizing energy F with respect to ϕ, the associated Euler-Lagrange equation for ϕ can be derived. Parameterizing the descent direction by an artificial time $t \geq 0$, the equation in $\phi(t, x, y)$ is [42]:

$$\frac{\partial \phi}{\partial t} = \delta(\phi)\left[\mu\, div\left(\frac{\nabla\phi}{|\nabla\phi|}\right) - v - \lambda_1(u_0 - c_1)^2 + \lambda_2(u_0 - c_2)^2\right] = 0 \quad (4.6)$$

In appendix I, the derivation of the whole equation is presented.

In figure 4.2 an example of Chan-Vese model segmentation result is illustrated. The first image shows the initial contour along the image border in red, and the second one shows the calculated contour along the object boundary after minimization. In this example, it shows that the Chan-Vese model offers good performance to detect objects or features with contours without gradients.

Indicated by this, it was investigated further for segmentation of cytoskeleton images.

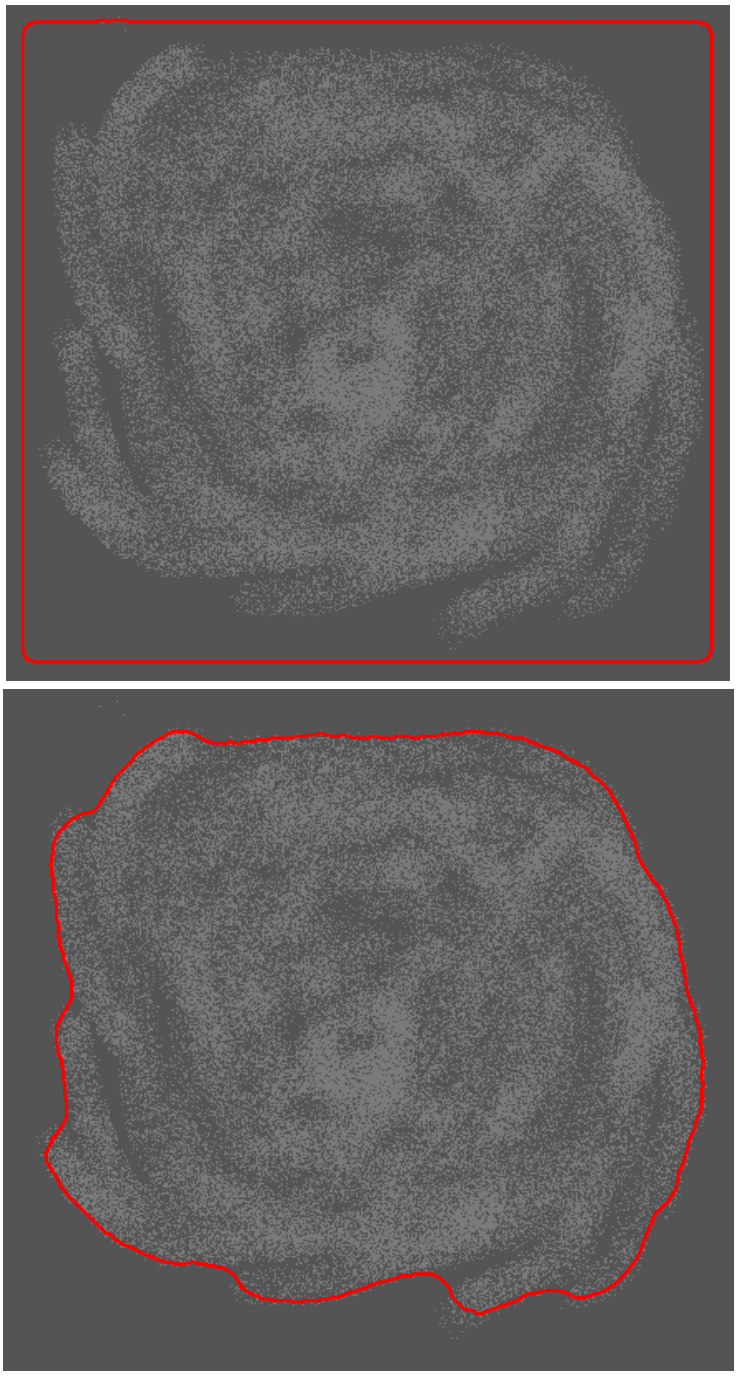


Figure 4.2 Program testing result of active contour model

4.1.2 Hybrid method

For the Chan-Vese model presented in the previous section, it is unfortunate that the model is not able to successfully accomplish image segmentation for some medical data for two reasons [43]. Firstly, lack of the boundary terms can lead to inaccurate segmentation for images with relatively well-defined boundaries. Secondly, the global nature of μ_{in} and μ_{out} can be a problem for images with complex backgrounds [43]. To overcome these two issues, the hybrid level-set based segmentation method uses both the object boundaries and region information to achieve robust and accurate segmentation results [43]. While the precise location of the target object can be detected with the aid of boundary information, the region information can be used to prevent boundary leakage.

The proposed function to be minimised in the hybrid method is defined as [43]:

$$F(\phi) = -\alpha \int_{\Omega} (I - \mu) H(\phi) d\Omega + \beta \int_{\Omega} g|\nabla H(\phi)| d\Omega \qquad (4.7)$$

where *I* represents the input image, *g* is the boundary feature map related to image gradient, Ω represents image domain, and α and β are predefined parameters to equilibrate these two terms. The first term of the function defines the region term where μ is a predefined parameter indicating the lower bound of the grey-level of the target object. In addition, the first term encourages the contours to enclose the regions with grey-levels greater than μ. The second term illustrates the geodesic active contour function, i.e. $\mathrm{F}(C) = \int g\big(C(p)\big)|C'(p)|dp$ [43], which is represented in the level-set formulation. The role of this term is to

encourage the contours to attach to the areas with high image gradients. Therefore, with the associated Euler-Lagrange equation, the following partial differential equation is obtained [43]:

$$\phi_t = |\nabla\phi| \left[\alpha(I-\mu) + \beta div\left(g\frac{\nabla\phi}{|\nabla\phi|}\right)\right] \tag{4.8}$$

In appendix II, the derivation details of this equation are presented.

In order to gain more insights about the model, using the identity $iv(g\vec{f}) = \langle\nabla g, \vec{f}\rangle + gdiv(\vec{f})$, where $\langle *, *\rangle$ denotes the inner product, the equation can be re-written as [43]:

$$\phi_t = |\nabla\phi| \left[\alpha(I-\mu) + \beta\,\langle\nabla g, \frac{\nabla\phi}{|\nabla\phi|}\rangle + \beta g div\left(\frac{\nabla\phi}{|\nabla\phi|}\right)\right] \tag{4.9}$$

Since $C_t = \gamma\vec{N}$ and $\phi_t = \gamma|\nabla\phi|$ describe the same curve evolution, the equation describing the explicit curve evolution is [43]:

$$C_t = \alpha(I-\mu)\vec{N} - \beta\langle\nabla g \cdot \vec{N}\rangle\vec{N} + \beta g\kappa\vec{N} \tag{4.10}$$

where the normal $\vec{N} = -\nabla\phi/|\nabla\phi|$ and the curvature $\kappa = div(\nabla\phi/|\nabla\phi|)$. If ϕ is a signed distance function (SDF), i.e. $|\nabla\phi| = 1$, then the minimised equation can be simplified to [43]:

$$\phi_t = \alpha(I-\mu) + \beta div(g\nabla\phi) \tag{4.11}$$

To test this method, an example of applying the hybrid method to a test image is illustrated in figure 4.3. This figure shows a successful segmentation result of a

multi-object image, where the first image shows the initial contour in a blue circle, and the second image shows the segmentation result.

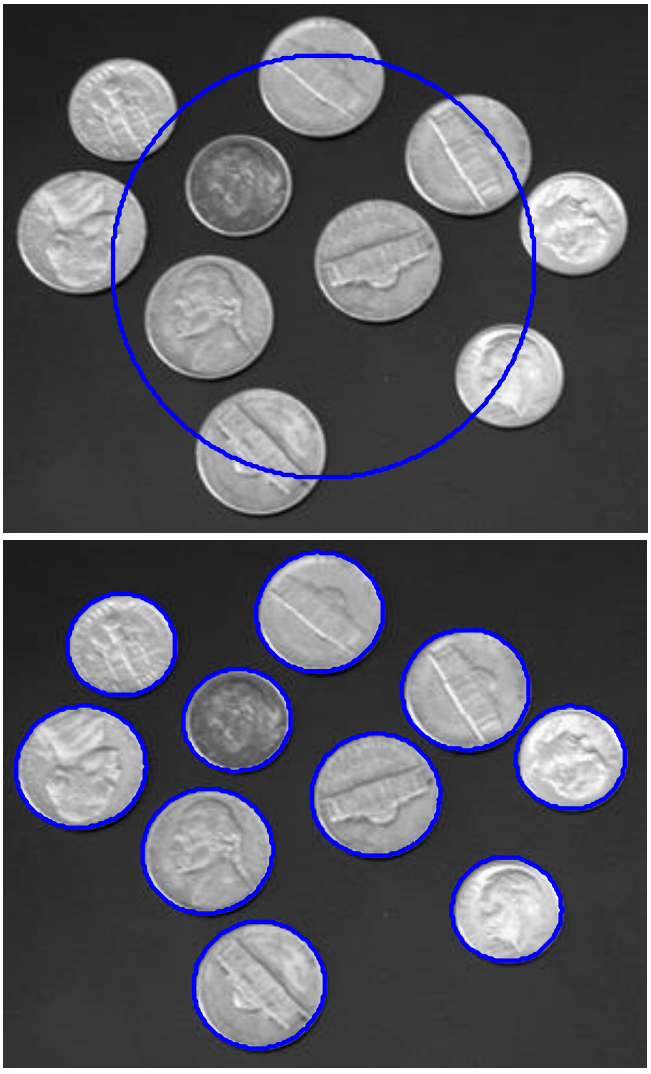


Figure 4.3 Test result of hybrid level-set method

4.1.3 Subjective surface model

The subjective surface method is a geometric model for segmentation of images with missing boundaries [44] [45]. The method is based on the mean curvature evolution of a graph with respect to the Riemannian metric induced by the image [46]. To apply this method to the provided images in this project, three processes

were run in sequence: image denoising, low level features extraction, i.e. edge detector, and segmentation using the subjective surface technique [47]. Each of these three parts is presented in the following sub-sections.

4.1.3.1 Image denoising

Since the noise present in the image can disrupt the shape information, denoising is an essential preliminary task in image segmentation. Image denoising can improve signal to noise ratio thereby enhancing object shapes, and define accurate position of the object boundaries. The filter named geodesic curvature filtering had been proved and suggested to be suitable for the dataset.

Along with the discussed noise reduction method, additional pre-processing of the dataset is also required. In order to achieve more accurate and reliable results, the rolling ball method for background subtraction and non-linear diffusion filter for noise reduction were performed.

4.1.3.2 Edge detector

The initial task of the segmentation strategy is to extract the so-called low level image features. For such purpose a classical solution is to consider an edge indicator $g = g(x, y, z)$, a smooth non-increasing function of the image gradient [47]

$$g(x, y, z) = \frac{1}{1 + (|G_\sigma(x, y, z) * I(x, y, z)|/\beta)^n} \tag{4.12}$$

where $G_\sigma(x, y, z)$ is a Gaussian kernel with standard deviation σ, * denotes the convolution, $I = I(x, y, z)$ represents the filtered imaged, and n is typically 1 or 2.

The parameter σ determines the minimal size of details that can be preserved, whereas β is related to the image grey levels mapped into the g function. The value of g is close to 1 in flat areas and close to 0 in the regions where image gradient is high. Thus, the minima of g denote the position of the edges and its minus gradient is a force field that can be used to drive the evolution, because it always points in the local edge direction.

4.1.3.3 Segmentation

In the subjective surface model, an initial function $\Phi_0 = \Phi_0(x, y, z)$ is constructed, in the image domain M, usually as a distance or peak function. Let $\Phi = \Phi(x, y, z, t)$ be an evolution of Φ_0, where t represents a synthetic time known in literature as a scale parameter and $\Phi(x, y, z, 0) = \Phi_0$. Φ is evolved afterward by a mean curvature motion to minimize the volume of the hypersurface S through the following motion equation [47]:

$$\Phi_t = H_g |\nabla\Phi|_a \tag{4.13}$$

where

$$H_g = gH + (\nabla g \cdot \nabla\Phi)\frac{1}{|\nabla\Phi|_a} \tag{4.14}$$

represents the mean curvature of S and [47]

$$|\nabla\Phi|_a = \sqrt{a + |\nabla\Phi|^2} \tag{4.15}$$

can be seen as a regularization of $|\nabla\Phi|$. In addition, H in the previous formula represents a mean curvature for S and it is given by [47]

$$\mathrm{H} = \frac{\left(a+\Phi_x^2+\Phi_y^2\right)\Phi_{zz} + (a+\Phi_x^2+\Phi_z^2)\Phi_{yy} + \left(a+\Phi_y^2+\Phi_z^2\right)\Phi_{xx}}{(a+\Phi_x^2+\Phi_y^2+\Phi_z^2)^{3/2}} - 2\frac{\Phi_x\Phi_z\Phi_{xz} + \Phi_x\Phi_y\Phi_{xy} + \Phi_y\Phi_z\Phi_{yz}}{\left(a+\Phi_x^2+\Phi_y^2+\Phi_z^2\right)^{\frac{3}{2}}} \quad (4.16)$$

Let $\Phi_t = H_g|\nabla\Phi|_a$ be represented in a more general formulation by adding two different weights, μ and ν, to the first and second term of the right side of the formula for H_g. By considering boundaries and initial conditions, the model equation can be written as follows [47]:

$$\begin{cases} \Phi_t = \mu g H|\nabla\Phi|_a + \nu\nabla g\cdot\nabla\Phi, & \text{in M} \times]0,\tau[\\ \Phi(x,y,z,t) = \min(\Phi_0), & in\ \partial M \times]0,\tau[\\ \Phi(x,y,z,0) = \Phi_0, & for\ (x,y,z) \in M \end{cases} \quad (4.17)$$

where τ is the value of scale parameter t which corresponds to the steady state condition for it. The parameter a is a stretching factor and represents a weight between two different dynamics. It indeed shifts the model from the mean curvature flow of level sets (a = 0) to the mean curvature flow of graph (a = 1). The first term on the right side of the equation represents a mean curvature H and weighted by the edge indicator g. The second term is a pure passive advection along the velocity field $-\nabla g$, whose direction and strength depend on the position. This term attracts the hypersurface in the direction of the image edges. Locally, different behaviours can be identified in the image regions according to one of these flows. The use of different weights between the regularization and the advective terms (ν>μ) facilitates the control of the evolution process. A higher weight of advective term ensures a better accumulation of image grey levels around existing contours.

To sum up, a test example of subjective surface model is presented in figure 4.4,where figure (a) is the original membrane image, figure (b) shows the segmented membrane contour in red and figure (c) shows surface S, constructed as a graph, at the end of the evolution with 10,000 scale steps.

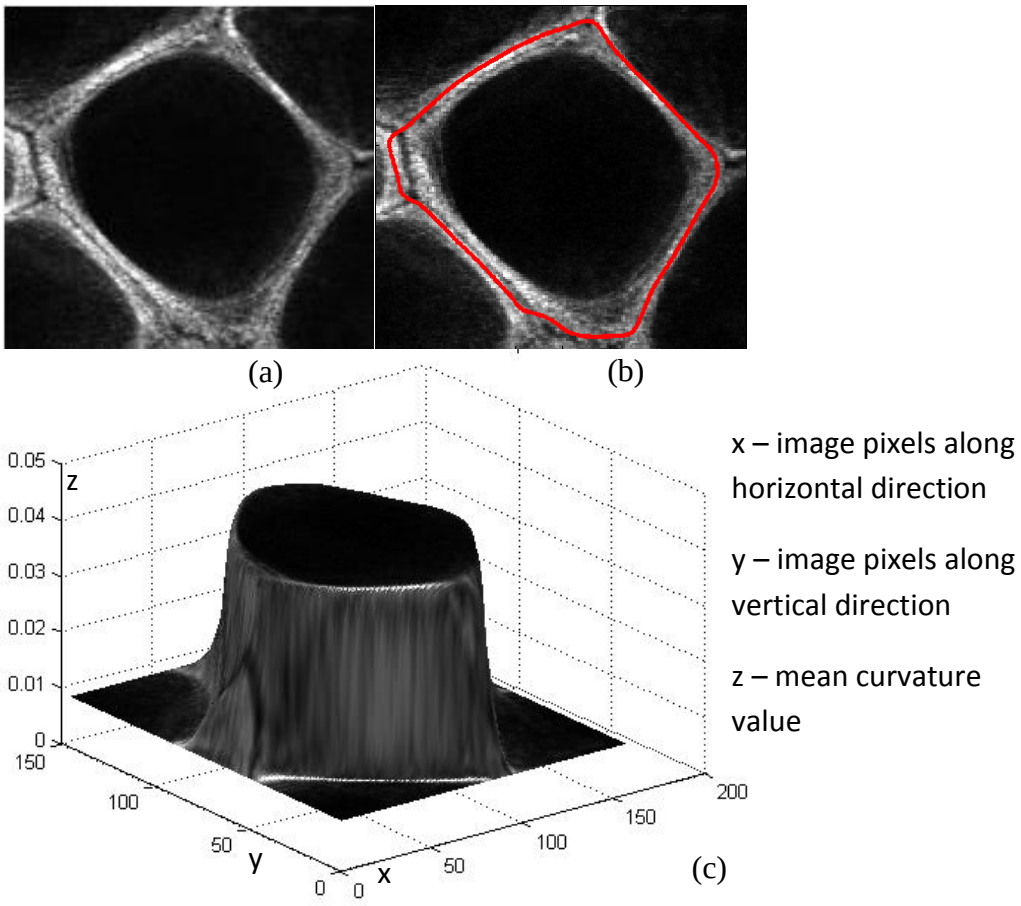


Figure 4.4 Example of membrane segmentation using subjective surface model

4.1.4 Numerical implementation

The common form of a level set function is written as $\phi_t = div(g\nabla\phi) + user\ designed\ components$ (4.18). The following numerical implementation shows the derivation of $div(g\nabla\phi)$.

Let the start point be denoted as

$$\frac{\partial \phi}{\partial t} = div(g \nabla \phi) \tag{4.19}$$

where $g \equiv g(|\nabla I|)$.

In one-dimension,

$$\frac{\partial \phi}{\partial t} \approx \frac{\phi_i^{k+1} - \phi_i^k}{\tau} \tag{4.20}$$

where τ represents the time sampling interval.

$$\frac{\partial}{\partial t}\left(g \frac{\partial}{\partial t}\phi\right) = \frac{\partial}{\partial t}\left(g \cdot \frac{\phi_{i+\frac{1}{2}} - \phi_{i-\frac{1}{2}}}{h}\right) = g_{i+\frac{1}{2}} \frac{\phi_{i+1} - \phi_i}{h^2} - g_{i-\frac{1}{2}} \frac{\phi_i - \phi_{i-1}}{h^2}$$

$$= \frac{g_{i+1} + g_i}{2} \cdot \frac{\phi_{i+1} - \phi_i}{h^2} + \frac{g_i + g_{i-1}}{2} \cdot \frac{\phi_i - \phi_{i-1}}{h^2} \tag{4.21}$$

where h represents the spatial sampling interval, $g_{i+\frac{1}{2}}$ and $g_{i-\frac{1}{2}}$ are calculated using linear approximation, $g_{i+\frac{1}{2}} = \frac{1}{2}(g_{i+1} + g_i),\ g_{i-\frac{1}{2}} = \frac{1}{2}(g_{i-1} + g_i)$.

Combining (4.20) and (4.21), and assuming that (4.21) is discretised at time "k" for an explicit scheme.

$$\frac{\phi_i^{k+1} - \phi_i^k}{\tau} = \sum_{j \in N(i)} \frac{g_j^k + g_i^k}{2h^2} \left(\phi_j^k - \phi_i^k\right) \tag{4.22}$$

where $N(i)$ represents neighbourhood of "i", in this case $N(i) = \{i - 1, i + 1\}$.

$$g_j^k = g_j(|\nabla I|) = g(|\nabla I(x_i)|) \tag{4.23}$$

Using a vector rotation

$$\phi^k = \left(\phi_1^k, \phi_2^k, \dots, \phi_i^k, \dots, \phi_M^k\right)^T \phi^k \tag{4.24}$$

$$\frac{\phi^{k+1} - \phi^k}{\tau} = A(\phi^k)\phi^k \tag{4.25}$$

where $A(\phi^k)$ is a matrix defined as:

$$A = \left[a_{ij}\right] \tag{4.26}$$

$$a_{ij} = \begin{cases} \dfrac{g_j^k + g_i^k}{2h^2} \; if \; j \in N(i) \\ -\displaystyle\sum_{n \in N(i)} \dfrac{g_j^k + g_n^k}{2h^2} \; if \; j = i \\ 0 \; else \end{cases} \tag{4.27}$$

Therefore the explicit iteration scheme in vector notation is given as:

$$\frac{\phi^{k+1} - \phi^k}{\tau} = A(\phi^k)\phi^k \tag{4.28}$$

or

$$\phi^{k+1} = \left(I + \tau A(\phi^k)\right)\phi^k \tag{4.29}$$

$$\begin{pmatrix} \phi_1 \\ \vdots \\ \phi_i \\ \vdots \\ \phi_N \end{pmatrix}^{k+1}$$

$$= \begin{bmatrix} -\dfrac{g_1 + g_2}{2h^2} & \dfrac{g_1 + g_2}{2h^2} & 0 & \cdots & \cdots & \cdots & \cdots & 0 & 0 \\ 0 & 0 & \vdots & \vdots & \vdots & \vdots & \vdots & 0 & 0 \\ 0 & \cdots & 0 & \dfrac{g_i + g_{i-1}}{2h^2} & -\dfrac{g_{i-1} + 2g_i + g_{i+1}}{2h^2} & \dfrac{g_i + g_{i+1}}{2h^2} & 0 & 0 & 0 \\ 0 & 0 & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & 0 & \cdots & \cdots & \cdots & 0 & \dfrac{g_{N-1} + g_N}{2h^2} & -\dfrac{g_{N-1} + g_N}{2h^2} \end{bmatrix}$$

$$* \begin{pmatrix} \phi_1 \\ \vdots \\ \phi_i \\ \vdots \\ \phi_N \end{pmatrix}^{k} \tag{4.30}$$

In this case g does not depends on “k”.

For the semi-implicit scheme, in equation (2) ϕ should be spatially discretised at time “k+1”, which gives

$$\frac{\phi^{k+1}-\phi^{k}}{\tau}=A(\phi^{k})\phi^{k+1} \tag{4.31}$$

or

$$\left(I-\tau A(\phi^{k})\right)\phi^{k+1}=\phi^{k} \tag{4.32}$$

For the implicit scheme, in equation (2) both ϕ and g should be spatially discretised at time “k+1”, which gives

$$\frac{\phi^{k+1}-\phi^{k}}{\tau}=A(\phi^{k+1})\phi^{k+1} \tag{4.33}$$

or

$$\left(I-\tau A(\phi^{k+1})\right)\phi^{k+1}=\phi^{k} \tag{4.34}$$

But in this case A does not depend on “k”.

The semi-implicit scheme requires solving a linear system, where the system matrix is tridiagonal and diagonally dominant. The most efficient way is to use the Thomas algorithm [48], a Gaussian elimination algorithm for tridiagonal systems. Assuming $Bu=d$ (4.35) is a tridiagonal linear system that needs to be solved, where

$$B = \begin{bmatrix} \alpha_1 & \beta_1 & & & \\ \gamma_1 & \alpha_1 & \beta_1 & & \\ & \ddots & \ddots & \ddots & \\ & & \gamma_{N-2} & \alpha_{N-1} & \beta_{N-1} \\ & & aa & \gamma_{N-1} & \alpha_N \end{bmatrix} \tag{4.36}$$

the Thomas algorithm contains three steps:

Step 1: LR decomposition

Decompose B into the product of a lower bidiagonal matrix L and an upper bidiagonal matrix R:

$$L = \begin{bmatrix} 1 & 0 & 0 & 0 \\ l_1 & 1 & 0 & 0 \\ 0 & \ddots & \ddots & 0 \\ 0 & 0 & l_{N-1} & 1 \end{bmatrix} \tag{4.37}$$

$$R = \begin{bmatrix} m_1 & r_1 & 0 & 0 \\ 0 & \ddots & \ddots & 0 \\ 0 & 0 & m_{N-1} & r_{N-1} \\ 0 & 0 & 0 & m_N \end{bmatrix} \tag{4.38}$$

Comparing the coefficients shows that $r_i = \beta_i$ for all i, and m_i and l_i can be obtained as follows:

$$m_1 = \alpha_1$$

$$For\ i = 1, 2, \dots, N-1$$

$$l_i = \frac{\gamma_i}{m_i} \tag{4.39}$$

$$m_{i+1} = \alpha_{i+1} - l_i \beta_i \tag{4.40}$$

Then $LRu = d$ (4.39) is solved in next two steps.

Step 2: forward substitution

Solving $Ly = d$ for y gives

$$y_1 = d_1$$

$$y_i = d_i - l_{i-1}y_{i-1}, for\ i = 2, 3, \dots, N \tag{4.42}$$

Step 3: backward substitution

Solving $Ru = y$ (4.43) for y gives

$$u_N = \frac{\gamma_N}{m_N} \tag{4.44}$$

$$u_i = \frac{y_i - \beta_i u_{i+1}}{m_i} \tag{4.45}$$

The Thomas algorithm is stable for every strictly diagonally dominant system matrix. To solve the linear system of the semi-implicit scheme in higher-dimensional cases, the additive operator splitting (AOS) scheme [48] will be used. The following algorithm shows the steps:

$input: u = u^n$

$v = K_\sigma * u$ after regularisation with a Gaussian convolution as pre-smoothing

Calculate diffusivity $g(|\nabla v^2|)$, where ∇v can be approximated by central differences.

Create copy $f = u$ and initialise the sum $u = 0$.

For $l = 1, \dots, m$, calculate $v = (mI - m^2 \tau A_l)^{-1} f$ and then solve N/N_l tridiagonal systems of size N_l by Thomas algorithm.

Then update $u = u + v$.

$output: u = u^{n+1}$.

The AOS schemes with a large time step can reveal average grey value invariance, stability based on extremum principle, Lyapunov functional, and convergence to a constant steady-state [48]. Therefore, AOS schemes are considered as an absolute discrete process rather than being intended to a continuous process.

4.2 Experimental results and discussion

The results of segmenting microfilament images contain two parts: cell nuclei segmentation and cell membrane segmentation. Both results were tested and shown in this section. In addition, some discussions and analysis of the results are also presented.

4.2.1 Investigation of parameter setting

In order to achieve the best possible results, several parameter sets are tested for all the methods. A coordinate system was designed to present results as a function of input parameters. In this way, the differences produced by changing the parameter sets can be illustrated clearly. Sensitivities of models to parameters can also be analysed.

4.2.1.1 Chan-Vese model

Firstly, for the Chan-Vese model, nine sets of parameters were chosen and for all the values of μ, λ_1 and λ_2, with the same time step Δt = 0.2. Figure 4.5 below illustrates the difference among different variables and how these variables influence the results.

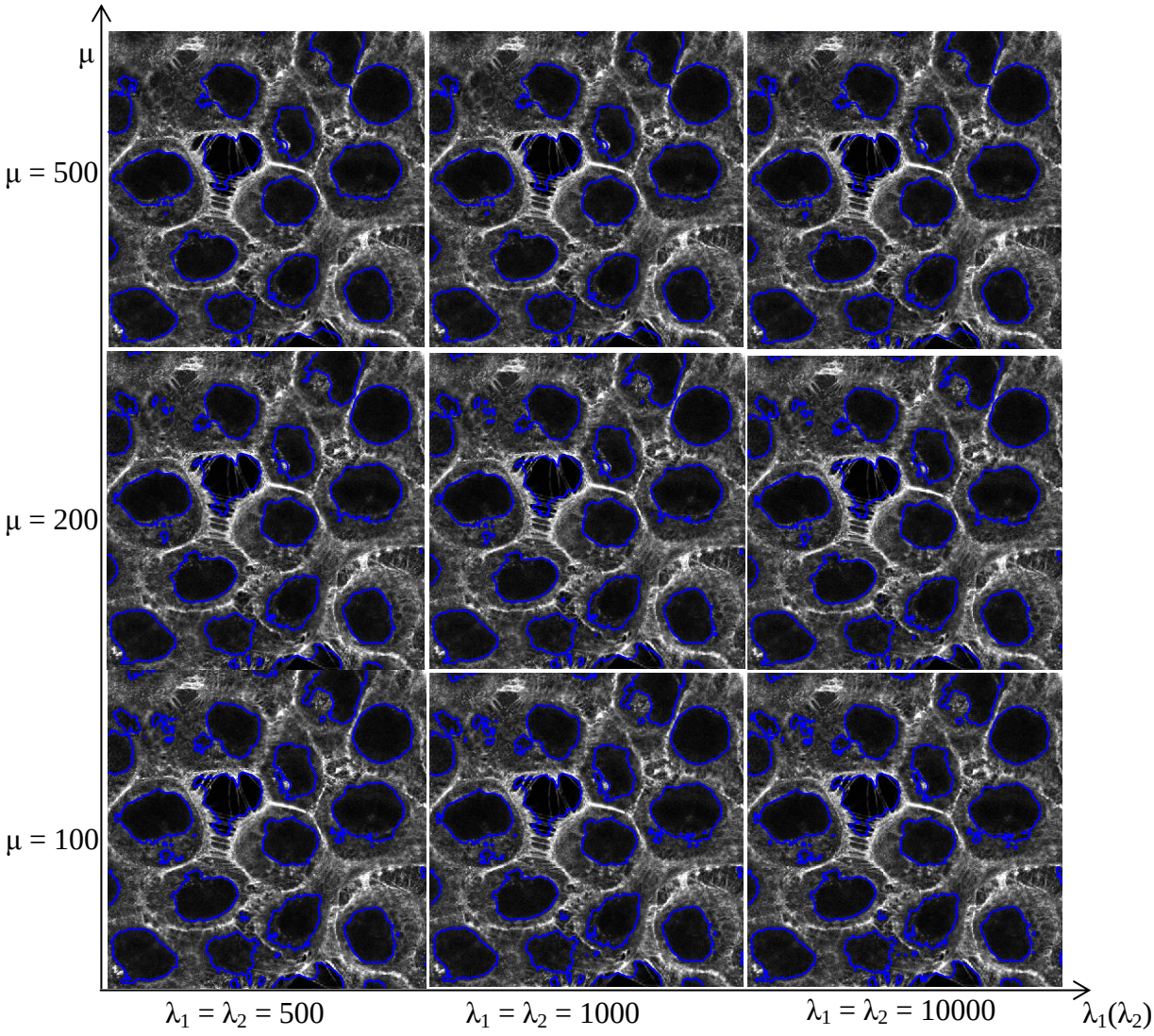


Figure 4.5 Output results by using Chan-Vese model of slice 35 in volume data

As the output results illustrated in figure 4.5, it is obvious to see that there are countable small redundancy contours in the results. By adjusting the variables, it is not difficult to find out that larger value of μ leads less redundancy output

contours. However, too large value of μ will mislead the method to consider two close nuclei as one nuclear, which causes unexpected errors in the results. For the variables λ_1 and λ_2, larger value will reduce the iteration time, which improves the computation efficiency.

4.2.1.2 Hybrid method

For the hybrid method, also nine sets of parameters were chosen, and for all values of α and β are with the same time step value Δt = 0.2.

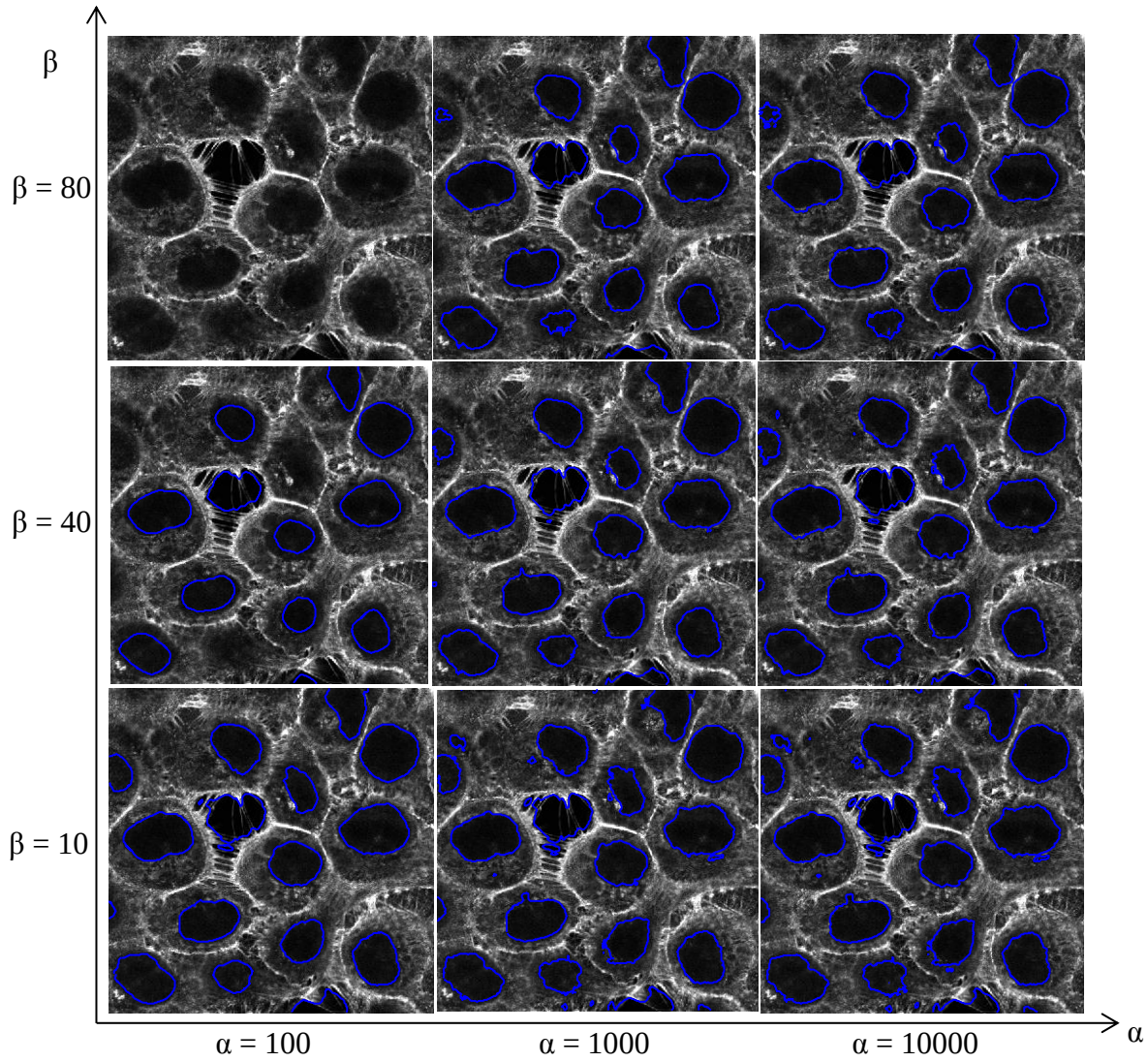


Figure 4.6 Output results by using hybrid method of slice 35 in volume data

In figure 4.6, results are shown with different parameters for using the hybrid method, and it can be observed that for a fixed value of α, larger value of β will induce less redundancy contours in the result. On the other hand, for a fixed value of β, the larger the value for α, the more segmentation contours, including some uninterested contours, appear. Similar as the Chan-Vese model results, larger β values help to reduce the iteration time. However, it does not mean that larger β values are better. By observing the result shown above, larger β values lead to disappearance of interested contours in the output.

4.2.1.3 Subjective surface model

In the segmentation part, there are two weights, μ and ν, for controlling the size of membrane segmentation area. In the results illustrating in figure 4.7, nine sets of different values for μ and ν with the same time step τ = 0.2 were chosen.

From figure 4.7, it can be seen that for a fixed value of parameter ν, the larger the value of μ is, the better the result produced. For a fixed value of μ, the larger ν value does not produce a better result. In the case of setting μ = 0.4, the result given by ν = 4 is even worse than the result given by ν = 2. Therefore, for an appropriate fixed ν, a larger value of μ may generally give a satisfactory membrane contour. To measure the accuracy of different parameter sets thereby determining the preferred parameter set, a comparative evaluation with respect to ground truth will be presented in the next section.

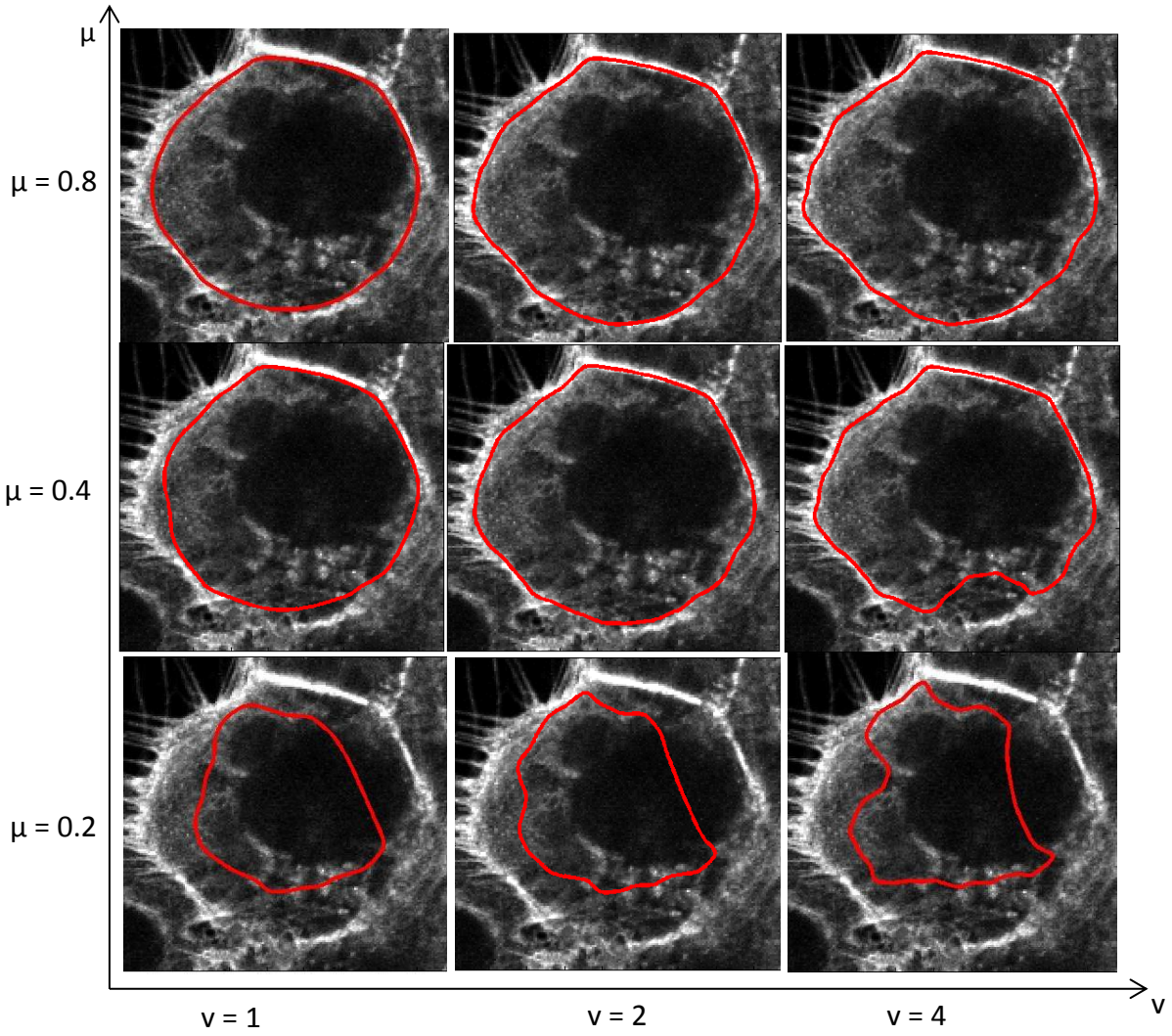


Figure 4.7 Membrane segmentation results of different weights by using subjective surface method

4.2.2 Investigation of segmentation performance

Ground truth is put forward to measure the accuracy of the segmentation results produced by the foregoing methods. Ground truth represents the expected segmentation results of the input image, and the ground truth of nuclei is shown in figure 4.8.

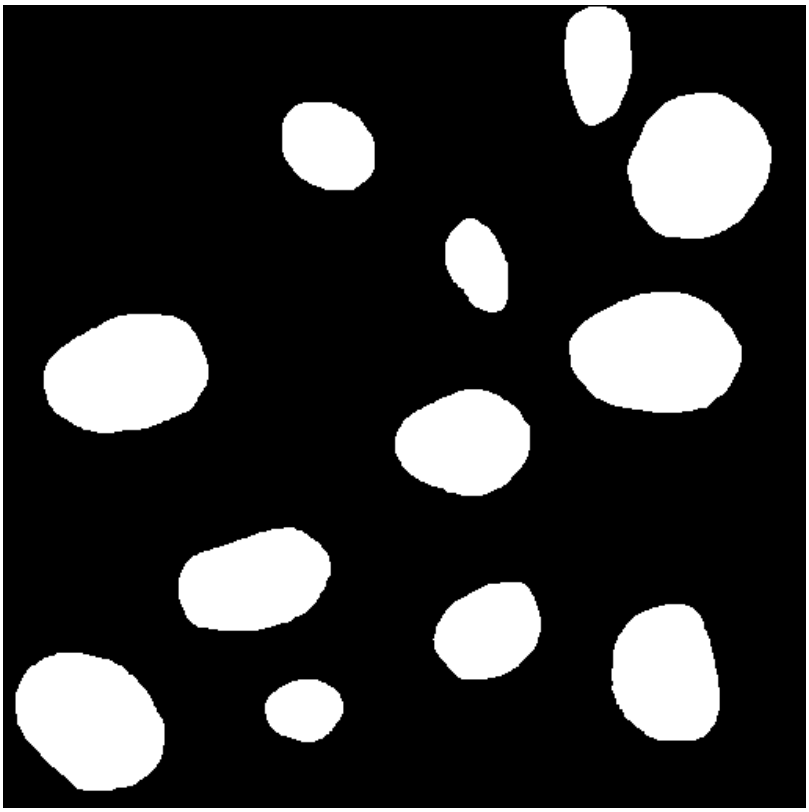


Figure 4.8 Ground truth of cell nuclei obtained from microfilament image

Denoting the ground truth and the contours produced as A and B, respectively. The segmentation accuracy can be calculated by $\frac{2(A \cap B)}{A \cup B}$ (4.44). Tables 4.1 and 4.2 illustrate the accuracy of Chan-Vese model and hybrid method.

μ \ $\lambda_1 = \lambda_2$	500	1000	10000
100	0.7327	0.7318	0.7322
200	0.7381	0.7365	0.7355
500	0.7314	0.7311	0.7298

Table 4.1 Segmentation accuracy of Chan-Vese model

In table 4.1, it can be seen that the accuracy of Chan-Vese model is higher when $\lambda_1 = \lambda_2 = 500$. As illustrated in figure 4.5, for a fixed value of μ, larger λ induces

more redundancy contours. Judging objectively by the accuracy calculated in table 4.1, the preferred parameter set is $\mu = 200$, $\lambda_1 = \lambda_2 = 500$. This parameter will be set to test more images in next section.

β \ α	100	1000	10000
10	0.8401	0.8130	0.8113
40	0.7698	0.8240	0.8213
80	0	0.8144	0.8183

Table 4.2 Segmentation accuracy of hybrid method

In all the results shown in table 4.2, the output with parameter set of $\alpha = 100$, $\beta = 10$ is chosen as the preferred parameter set. Though a smaller value of β may lead to more time-consuming computation, the quality of the output makes the parameter set become the most favourable choice.

To measure the accuracy of segmentation produced by the subjective surface model, the ground truth of a one-cell membrane is given as shown in figure 4.9.

Figure 4.9 Ground truth of one-cell membrane obtained from microfilament image

Table 4.3 illustrates the segmentation accuracy produced by different parameter sets of the subjective surface model. It can be observed that the highest accuracy is produced by setting $\mu = 0.8$, $v = 4$. From the previous segmented images, it can be easily seen that this parameter set had given a good result. Therefore, parameter set of $\mu = 0.8$, $v = 4$ is chosen to be the most favourable value.

μ \ v	1	2	4	6	8
0.2	0.6688	0.6934	0.6845	0.5820	0.5581
0.4	0.9130	0.9499	0.9321	0.9193	0.9156
0.6	0.9288	0.9665	0.9702	0.9594	0.9492
0.8	0.9475	0.9676	0.9762	0.9751	0.9716
1	0.9196	0.9617	0.9668	0.9645	0.9607

Table 4.3 Segmentation accuracy of subjective surface model

4.2.3 Experimental results

The Chan-Vese model with the preferred parameter sets determined in the previous section was applied to other images. These images are from same data volume as figure 4.5 but from different slices. Figure 4.10 shows the segmentation results of these slices by using the Chan-Vese model. Subfigures (a), (b), (c) and (d) represent the segmentation results of image slices 25, 30, 40 and 45, respectively.

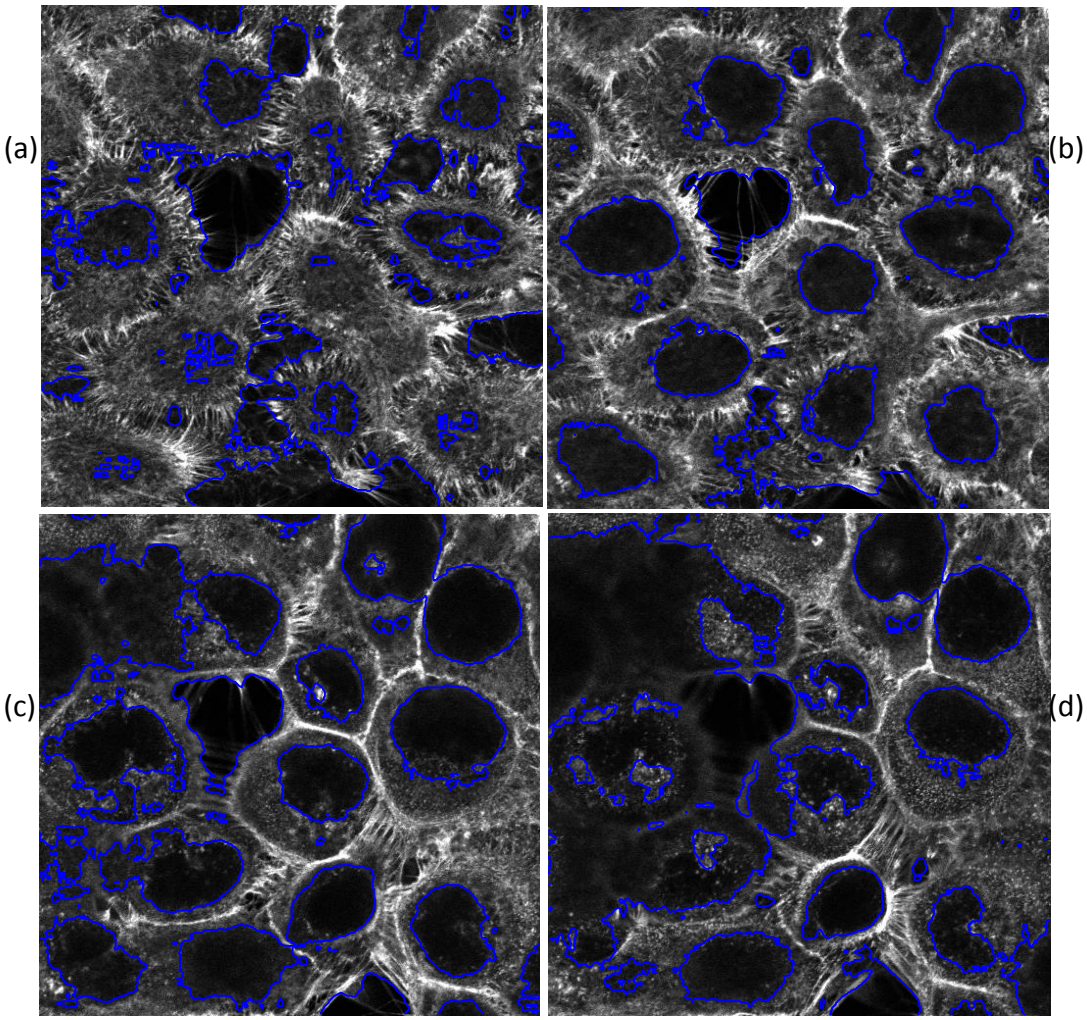


Figure 4.10 Results of Chan-Vese model applied to other images

Similar as the process of applying the Chan-Vese model to those images, the hybrid method was also applied to test the same images. Figure 4.11 shows the segmentation results of the same slices using the hybrid method. Subfigure (a), (b), (c) and (d) represent results of image slices 25, 30, 40 and 45, respectively.

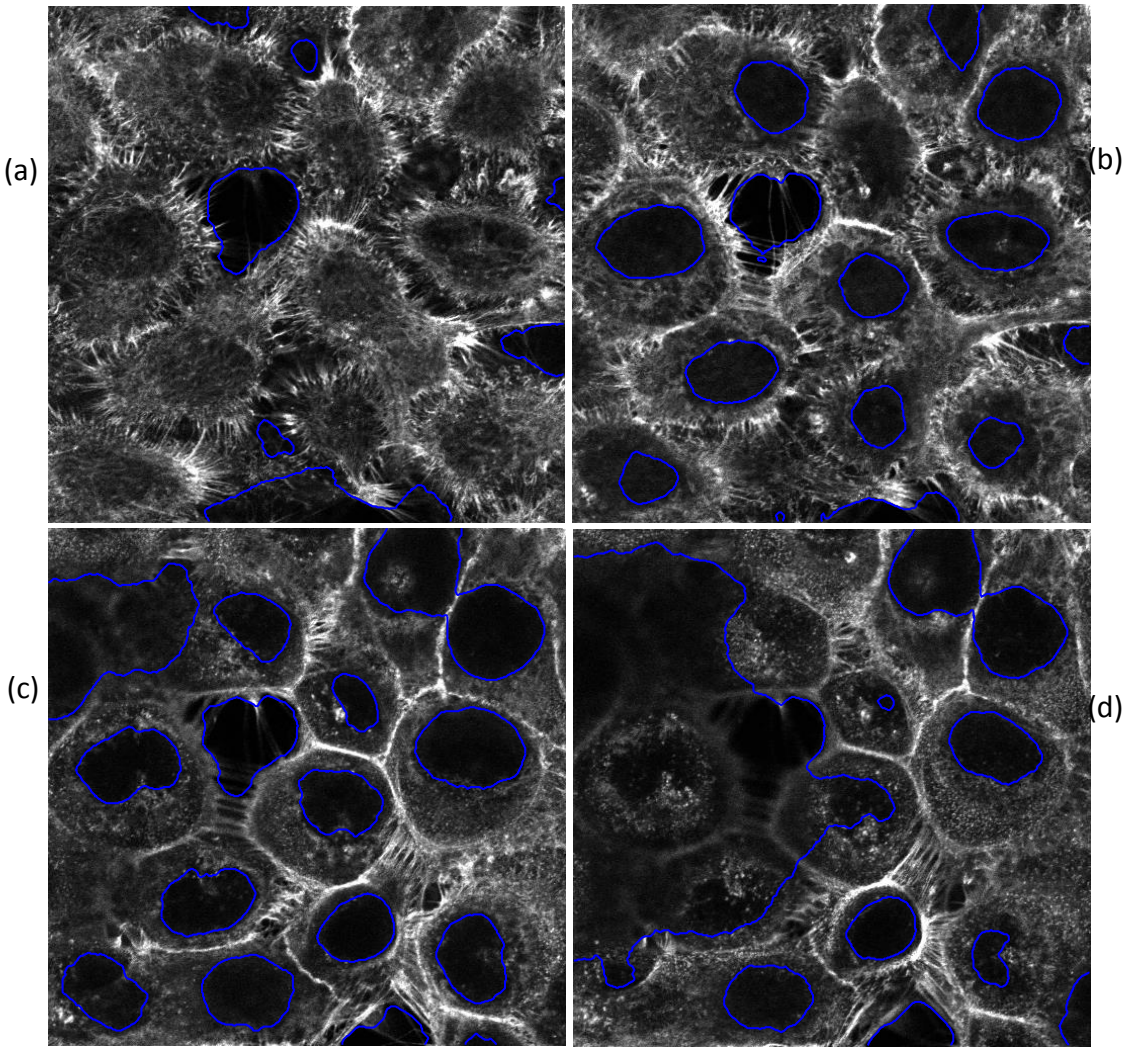


Figure 4.11 Results of hybrid method applied to other images

There are eight cells in the test image slice. Figure 4.12 presents all the membrane segmentation results of every cell in that slice. Figure 4.12a shows the original data and figure 4.12b shows all the segmentation results.

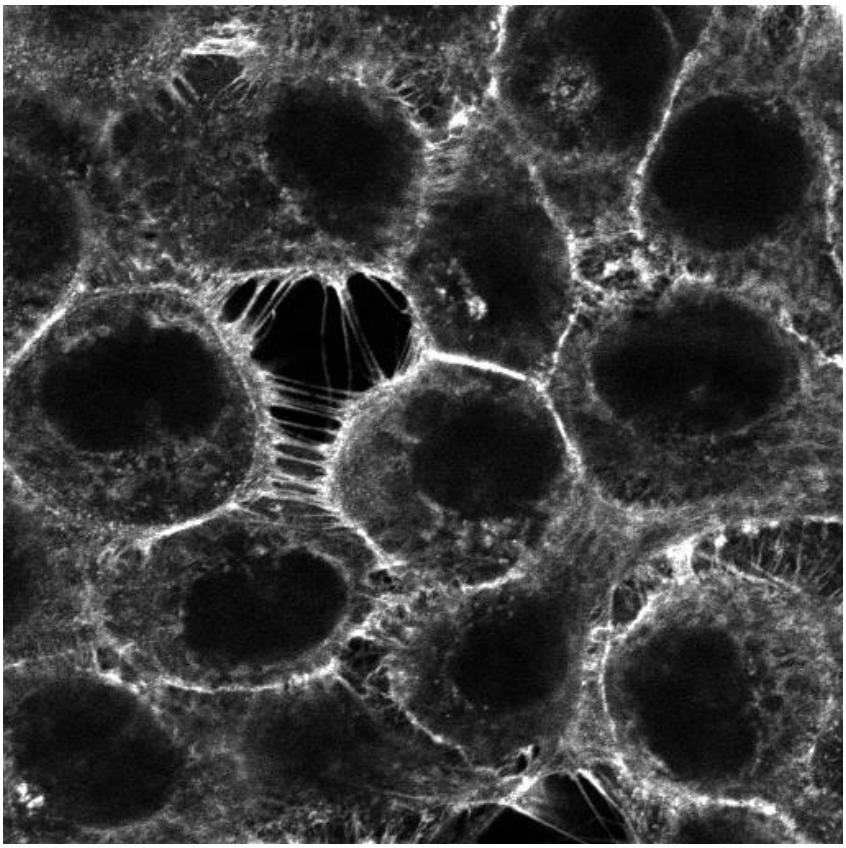
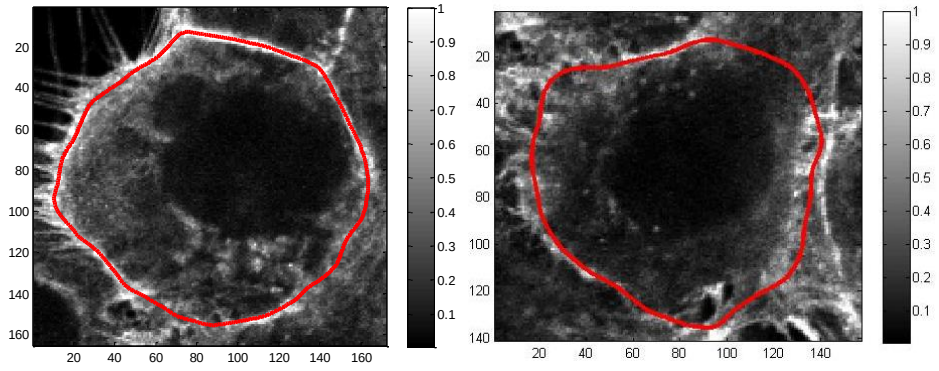
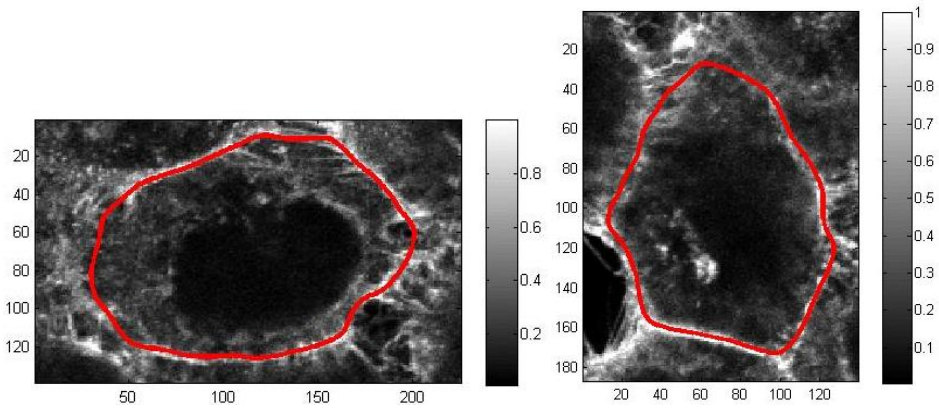


Figure 4.12a Initial input data image





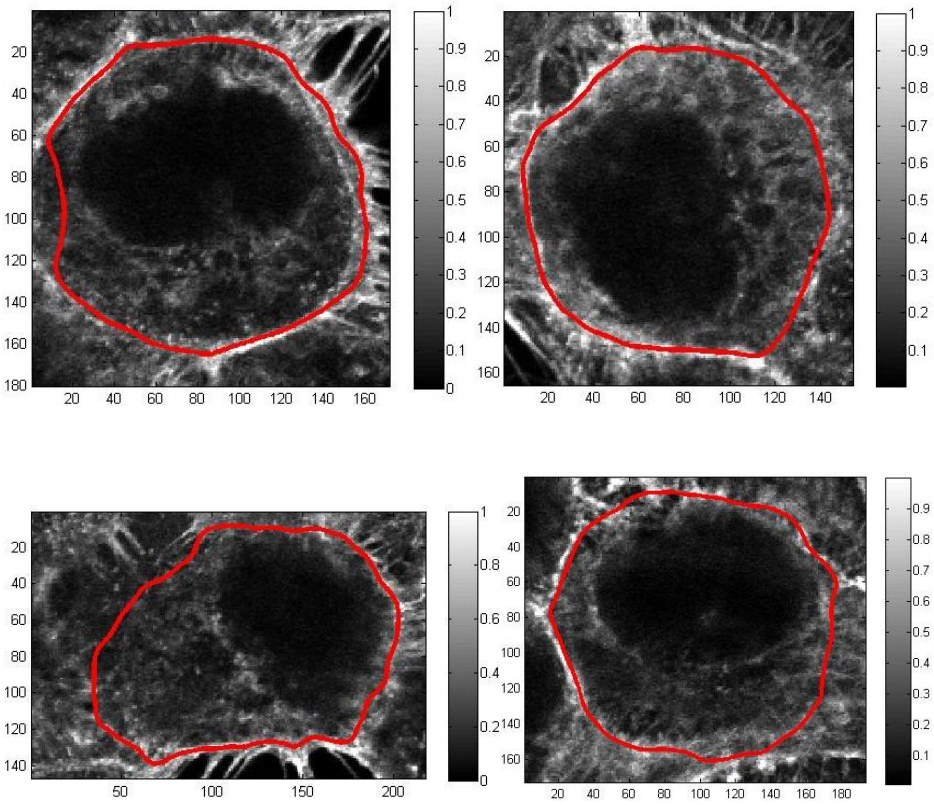


Figure 4.12b Segmentation results of every effective cell from the input data image

4.2.4 Discussion

From the accuracy illustrated in tables 4.1 and 4.2, it can be observed that hybrid method produces better segmentation results than the Chan-Vese model. In order to evaluate the time consumed by both the Chan-Vese model and the hybrid method, computation time of different runs was tabulated in tables 4.4 and 4.5.

	Computation time (Chan-Vese model), second		
	μ = 100, λ = 500	μ = 200, λ = 1000	μ = 500, λ = 10000
1st testing	41.454378	44.414945	56.572380
2nd testing	40.891650	46.452894	56.084843
3rd testing	37.407635	47.758451	55.698905
4th testing	41.081127	47.851681	55.968173
5th testing	41.297630	49.398942	55.092346
6th testing	41.926855	47.531497	56.867782
7th testing	40.683410	47.906111	56.521856
8th testing	40.787470	47.121221	53.834669
9th testing	41.119005	46.429717	55.757780
10th testing	41.470418	46.348638	55.305817
Average	40.811958	47.121410	55.770455

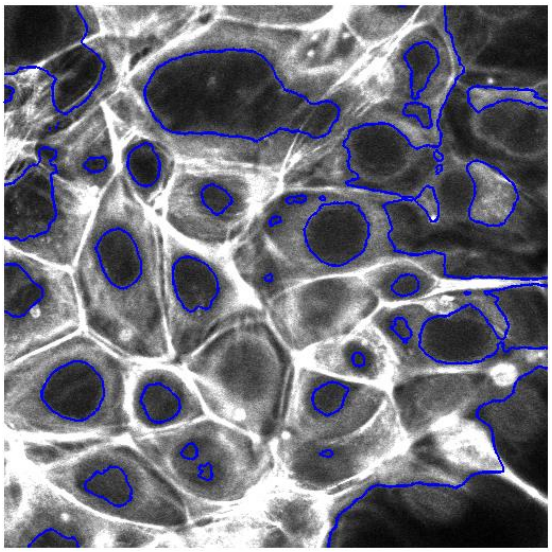
Table 4.4 Computation time of Chan-Vese model

	Computation time (hybrid method), second		
	α = 100, β = 10	α = 1000, β = 40	α = 10000, β = 80
1st testing	14.151760	14.723267	13.288495
2nd testing	13.192585	14.212260	13.463682
3rd testing	13.127061	14.287021	13.220852
4th testing	12.707522	14.354674	13.251232
5th testing	13.266632	14.832933	13.280893
6th testing	13.437330	14.478705	13.245812
7th testing	13.011324	14.200345	13.347734
8th testing	13.182254	14.407971	13.479779
9th testing	13.161008	14.267131	13.307790
10th testing	13.550061	14.377400	11.936549
Average	13.278754	14.414171	13.182282

Table 4.5 Computation time of hybrid method

For these two tables above, all the testing results were acquired using Matlab 2008 under 32-bit windows 7 system. From the listed tables, it can be observed that the Chan-Vese model is much more time-consuming than the hybrid method.

Chan-Vese model



Hybrid method

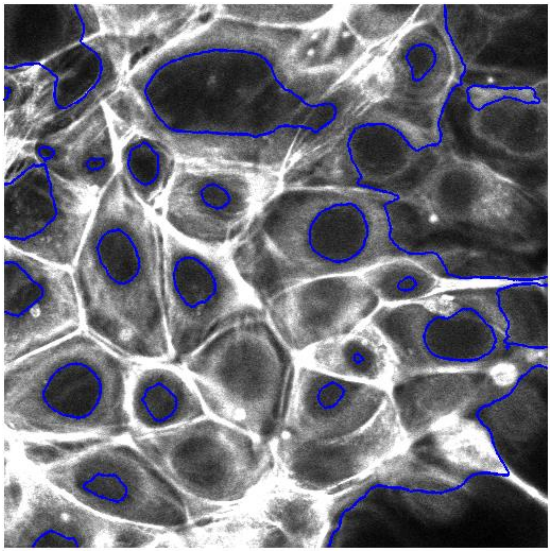
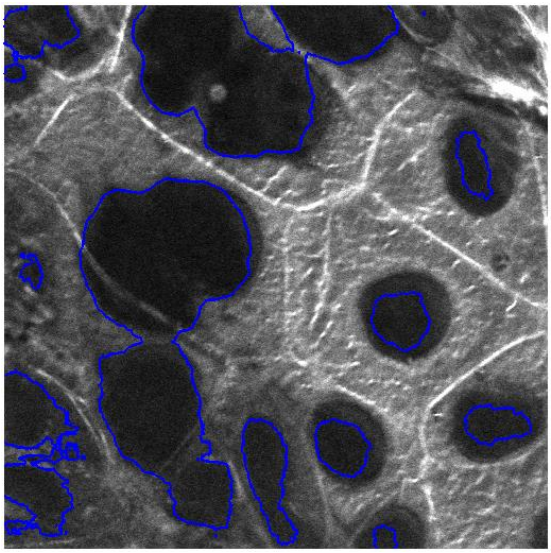


Figure 4.13 Comparison of Chan-Vese model and hybrid method results (1)

Chan-Vese model



Hybrid method

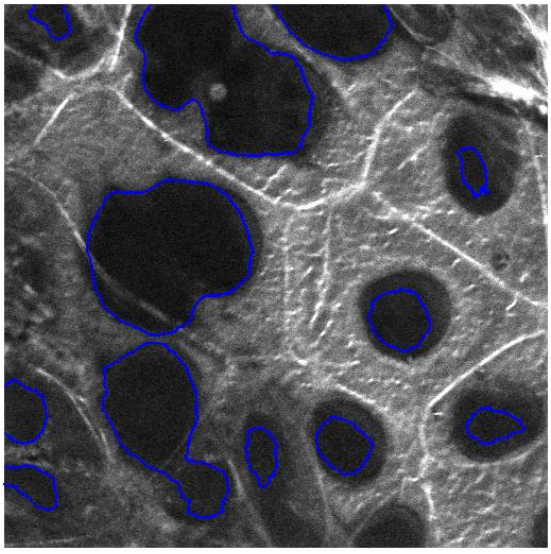


Figure 4.14 Comparison of Chan-Vese model and hybrid method results (2)

In figure 4.13 and figure 4.14, two more images are tested using both the Chan-Vese model and the hybrid method. For both of these two figures, the result obtained using the Chan-Vese model are shown on the top and the result obtained using the hybrid method are at the bottom. In addition, considering all the results and comparison achieved in the former sections, it is obvious that the

Chan-Vese model creates more redundancy contours than the hybrid method does. And the Chan-Vese model has easier treatment to two close nuclei as one while hybrid method will not. It is evident that the hybrid method had demonstrated more accurate results compared with the Chan-Vese model for these microfilament data.

4.2.5 Nuclei segmentation in 3D

From comparison and analysis of the results obtained by using the Chan-Vese model and the hybrid method in the above sections, the hybrid method has been proved to be more reliable to obtain more accurate segmentation results. Therefore, in this three dimensional stage, the hybrid method will be used for 3-D nuclei segmentation.

Because of the time-consuming computation for 3-D segmentation, a volume data with a small number of slices has been tested first. In this section, figure 4.15a in the following shows the 3-D results of 20-slice volume data (from slice 21 to slice 40) with the parameter setting of $\alpha = 100$, $\beta = 20$ and time step $\Delta t = 0.2$. In order to see the detail in the results, figure 4.15b shows how the segmentation taking place in ten different slices of the data.

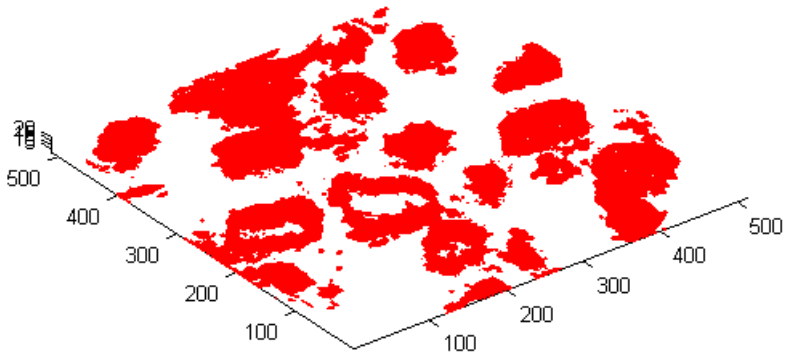


Figure 4.15a 3-D nuclei segmentation of 20-slice volume data

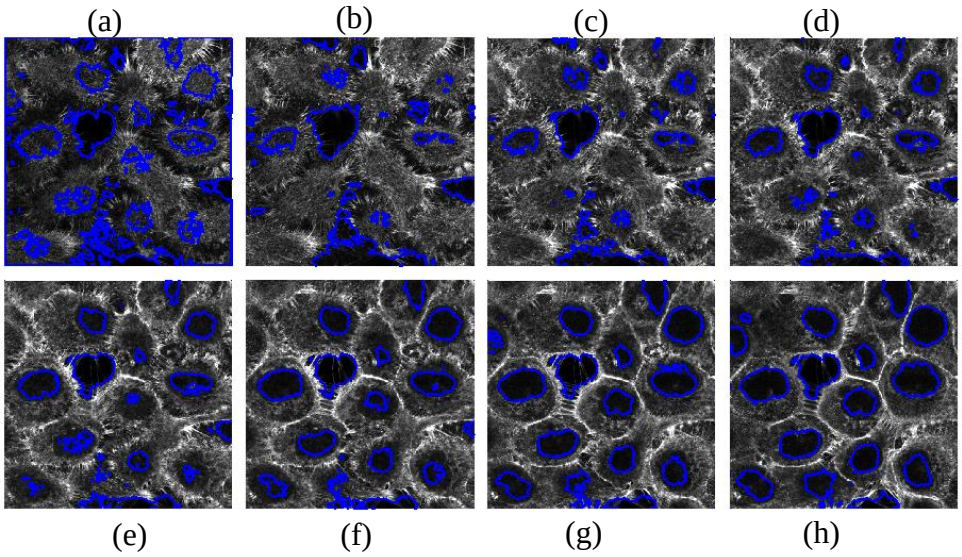


Figure 4.15b 3-D nuclei segmentation results in different slices. (a) slice 21; (b) slice 23; (c) slice 26; (d) slice 28; (e) slice 30; (f) slice 33; (g) slice 36; (h) slice 39.

In figure 4.15a, the result is not displayed clearly and distinctly enough in the 3-D case. The reason is that not all slices are contained in the volume data due to the limitation of computational memory. This restriction indicates the shortage of this algorithm dealing with 3-D microfilaments data and this is a potential area for improvement in the future work.

In order to make the 3-D view more obvious under the case of restricted computational memory, the original data are resized into smaller size (64×64) and the new volume data contains 22 slices of resized cell images. The parameters are set as $\alpha = 100$, $\beta = 1$ with time step value $\Delta t = 0.2$. Figure 4.16a below illustrates the 3-D nuclei segmentation result, which is clearer comparing to the result shown in figure 4.15a.

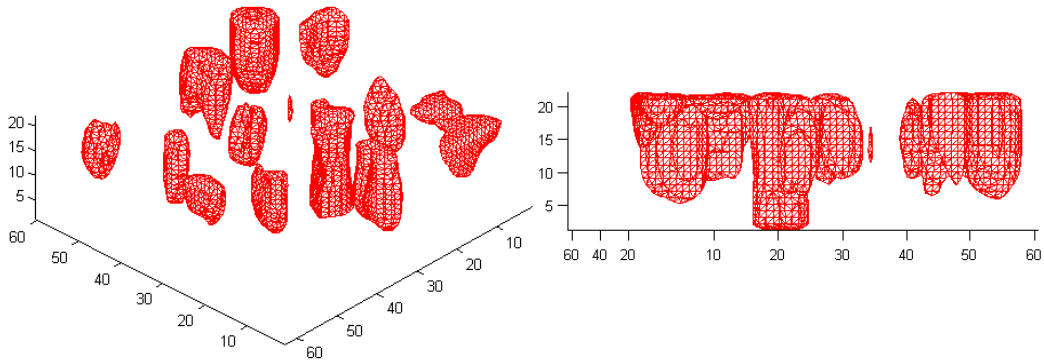


Figure 4.16a Modified 3-D nuclei segmentation of 22-slice volume data

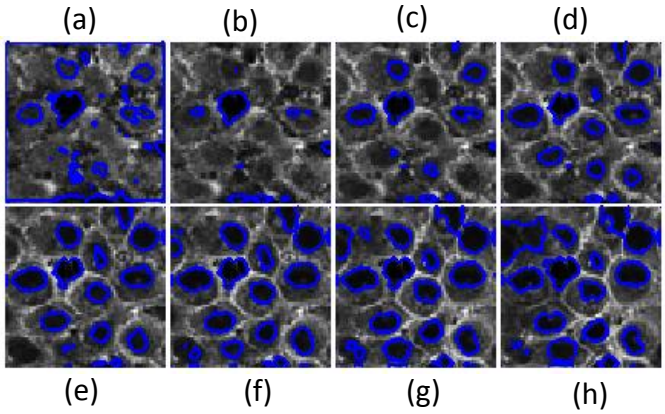


Figure 4.16b Modified 3-D nuclei segmentation results in different slices. (a) slice 21; (b) slice 23; (c) slice 26; (d) slice 28; (e) slice 30; (f) slice 33; (g) slice 36; (h) slice 39.

Compared to figure 4.15a, this 3-D result is more distinct and the redundant segmented volume is less. Although it causes data information loss after shrinking the images, this action helps to visualise a better 3-D nuclei segmentation result using the hybrid method.

To evaluate the 3D segmentation results, the accuracy values are calculated and listed in table 4.6. The comparison of accuracies between using 2D hybrid method and 3D method is illustrated in table 4.6 and figure 4.17.

	2D hybrid method	3D hybrid method
Slice 21	0	0.0045
Slice 22	0.0037	0.0167
Slice 23	0.0103	0.0206
Slice 24	0.0296	0.0853
Slice 25	0.1164	0.1607
Slice 26	0.2738	0.3166
Slice 27	0.3865	0.4082
Slice 28	0.5079	0.5305
Slice 29	0.6521	0.6122
Slice 30	0.7406	0.6874
Slice 31	0.7826	0.7314
Slice 32	0.8129	0.7511
Slice 33	0.8103	0.7359
Slice 34	0.8053	0.7268
Slice 35	0.8019	0.7664
Slice 36	0.7926	0.7645
Slice 37	0.7646	0.7469
Slice 38	0.7313	0.7234
Slice 39	0.6952	0.6977
Slice 40	0.6583	0.6786

Table 4.6 Accuracy comparison of 2D and 3D hybrid method results

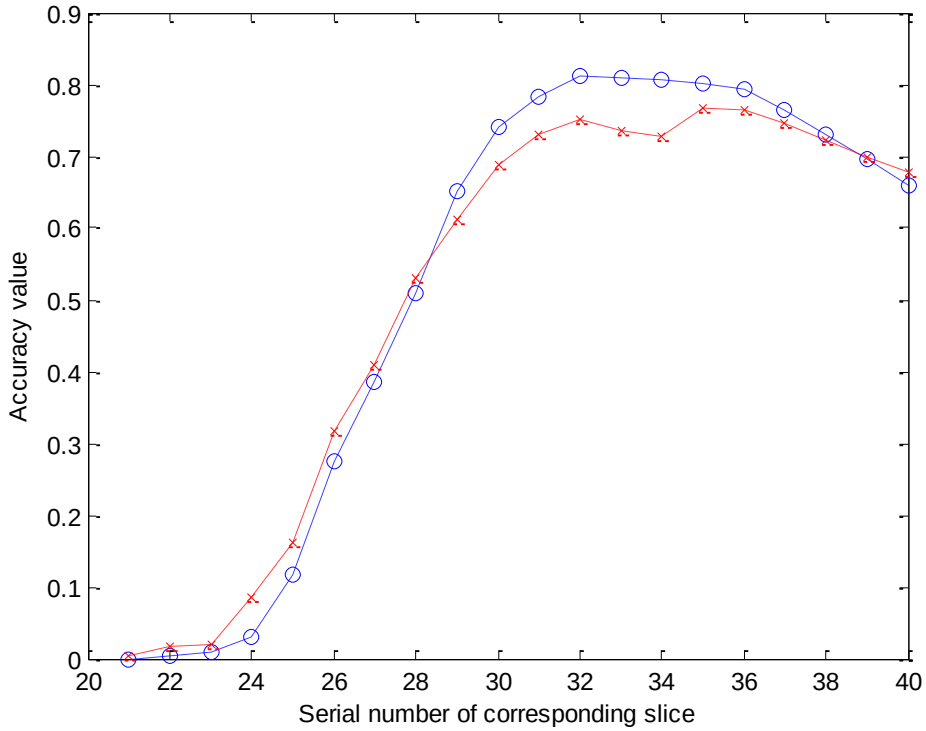


Figure 4.17 Accuracy values for both 2D and 3D hybrid method

By the information obtained from table 4.6, it can be observed that 3D hybrid method produces better results for first eight and last two slices. The tested data is parts of the 98 slices volume data containing 20 slices (from slice 21 to slice 40). Different from process each slice individually in 2D case, 3D hybrid method produces segmentations by concerning all the slices in the tested volume data. 2D hybrid method gives better results for the slices for the middle slices, but the accuracy value of 2D method for the first slice is zero. These low accuracy results reveals that both 2D and 3D method still have potential improvements, which may realised by combining with some other possible algorithms. Apart from that reason, there is another possibility may lead the segmentation results to low accuracy value. It is possible that nuclei do not exist in the earlier slices, or exist with relatively very small region. In this situation, the segmentation accuracy will turn to low value, or even be zero. In addition, the time consumption of 3D

segmentation is large as it requires large computation process. Consider of all the stated points, it cannot be definitely concluded that 3D model segment more accurately than 2D model. It is more equitable to be judged by users with consideration of related applications. These problems can be aimed to a potential research direction in the future work.

4.3 Concluding remarks

Microfilament data are normally composed of complex protein with flexible bended directions. Microfilament segmentation contains two main parts: cell nuclei segmentation and cell membranes segmentation. To realise cell nuclei and membranes segmentation, level set method and subjective surfaces model are applied, respectively. In addition, two different models, Chan-Vese model and hybrid method, are tested and compared in cell nuclei segmentation, and both these model manage to segment cell nuclei. By the comparison of segmentation accuracy values of these two level set models, hybrid method is considered as preferred model for nuclei segmentation. In cell membranes segmentation part, subjective surfaces model successfully segment cells with missing boundaries. With the aid of the foregoing segmentations as pre-processing, the biologist can carry on with some further studies of microfilament structures, such as cells counting, cells tracking, cells moving tendency prediction, *et al*. It may also be possible to carry out the analysis of cell's canceration prediction with more comprehending about microfilament data properties. These further studies make

accurate microfilament segmentation become so important. Furthermore, 3D cell segmentation can be considered as a better way to help observe and analysis cell structure. Therefore, to improve appropriate and accurate method for 3D microfilament segmentation can be considered as a potential future direction.

Chapter five

Conclusion and future work

This research aims to investigate and evaluate reliable segmentation methods for cytoskeleton images. Two types of datasets were used in this study, microtubule images and microfilament images. A particular focus has been placed on segmentation of microfilament images.

A multi-templates method was used to segment microtubule images. This method categorises the fibre constitutions in the data and marks fibre directions. It was applied to the raw data after eliminating background noises. This analysis quantifies the cytoskeletal features, so that it can offer reliable assays to evaluate cells with effects of drugs and other perturbations. Comparing with other algorithms, this method can obtain reliable results from the images, which are blurred due to defocus or contain high level background noise. It is applicable to both immunolabeled cells data and fluorescently tagged cytoskeletal proteins. However, the disadvantage of this method is long computation time due to complex correlation calculations. In addition, the orientations marked on the microtubule fibre constitution are related to the size of the correlation mask. For

larger correlation mask size, the marked orientation will become more accurate, but the computational time will be increased.

Investigations of microfilament images included cell nuclei segmentation and cell membrane segmentation. The approach based on active contours without edges (Chan-Vese model) was used to segment cell nuclei. With the help of the hybrid method, the results were improved. The subjective surface model was implemented for cell membrane segmentation. The most significant advantage of the Chan-Vese model is that it can detect objects whose boundaries are not defined by gradient. However, the Chan-Vese model showed poor performance on microfilament images because of missing boundaries. This problem was tackled by using the hybrid method. Therefore, the hybrid method is considered as a preferred method to segment cell nuclei. An unsolved problem in this study is to eliminate unnecessary segmentation of non-cell regions. The algorithm cannot identify gaps between separate cells. Another disadvantage is the merged contour problem. The algorithm sometimes merges two close contours and recognises them as one.

The subjective surface model was used to segment cell membranes. This algorithm realised membrane segmentation accurately in the case of missing boundaries. But this model is semi-automatic, as it needs manual selection of a central point and region of every cell. Another shortcoming is that it can only process one cell in one computation process.

As discussed above, several potential future extensions of this research are listed below.

Future work

a) Improve the multi-template method to mark fibre orientation more accurately and computation efficiency. A larger mask may be used to increase the accuracy, if the program could be written in C or similar high efficiency computer programming languages.
b) Modify the hybrid method for nuclei segmentation to avoid segmenting non-cell regions. A possible way to improve it is to add a shape prior section for segmentation results. Most regular cells are circular or ellipse in their shape. If the segmented contours were not in the similar shape, the corresponding area could be considered as non-cell regions.
c) Enhance nuclei segmentation methods to avoid merging area. A preferred way to solve this is to combine multiphase active contour segmentation with the hybrid method. Multiphase segmentation supports preservation of prior topological configuration between different objects, at the same time, it can provide soft shape constraints.
d) Improve the three dimensional program. Currently, it is only able to process 3D microfilament data with 20 slices, and the computer runs out of memory for processing the whole volume data with 98 slices. A potential future work is to focus on implementing three-dimensional program to solve such a time-consuming problem.

e) The subjective surface model can be improved to a fully automatic model. It is possible to add automatic selection of cell centre and region. In this case, this model can be improved to solve the segmentation problem of multiple cells.

References

[1] S. C. Rastogi (2004) *Cell Biology (Second edition)*. New Adge International Publishers. Page 248.

[2] B. J. Matuszewski, M. F. Murphy, D. R. Burton, T. E. Marchant, C.J. Moore, A. Histace, F. Precioso (2011) *Segmentation of cellular structures in actin tagged fluorescence confocal microscopy images*. Image Processing (ICIP), 2011 18th IEEE International Conference. Pages 3081 – 3084.

[3] Linda G. Shapiro and George C. Stockman (2001) *Computer Vision*. New Jersey, Prentice-Hall, Pages 279-325.

[4] X. Zhou (2007) *Study of image segmentation methods and their applications*. China Academic Journal Electronic Publishing House. Information Technology, Pages 1 – 3.

[5] B. Tan (2006) *Investigation and applications of image segmentation*. academic dissertation, Northeastern University, China. Changchun Institute of Technology Library, China. Pages 4 – 5.

[6] Z. Wu and R. Leahy (1993) *An optimal graph theoretic approach to data clustering: Theory and its application to image segmentation*. IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 15, Issue 11, Pages 1101 – 1113.

[7] Pham L. Dzung, Chenyang Xu, Jerry L. Prince (2000). *Current Methods in Medical Image Segmentation*. Annual Review of Biomedical Engineering, Vol. 2, Pages 315 – 337.

[8] Cheng Y., Hartemink C.H, Hartwig J.H., Forbes D.C. Jr. (1999) *Three-dimensional reconstruction of the actin cytoskeleton from stereo images*. J. Biomech, Vol. 33, Pages 105 – 113.

[9] N. Lichtenstein, B. Geiger, and Z. Kam (2002) *Quantitative Analysis of Cytoskeletal Organisation by Digital Fluorescent Microscopy*. Weizmann Institute of Science, Department of Molecular Cell Biology, Rehovot, Israel. Cytometry A. Vol. 54, Pages 8 – 18.

[10] Sternberg S. (1983). *Biomedical image processing*. Computer, Vol. 16, Issue 1. Pages 22 – 34.

[11] Hames, B. D. (2005) *Biochemistry (Third edition)*. Abingdon: Taylor & Francis, Page 149 – 162.

[12] Richard B. Vallee. Pt.B. *(1998) Molecular motors and the cytoskeleton*. Academic Press, Vol. 298, Pages 181 – 183.

[13] Fulton A. (1984) *The cytoskeleton: cellular architecture and choreography*. Cambridge and Hall, Page 7.

[14] I.I. Smalyukh, S.V. Shiyanovskii, O.D. Lavrentovich (2001) *Three dimensional imaging of orientational order by fluorescence confocal polarizing microscopy*. Chemical Physics Letters. Vol. 336, Issues 1–2, Pages 88 – 96.

[15] J. B. Pawley (2006) *Handbook of Biological Confocal Microscopy (Third edition)*. Springer Science, Page 368.

[16] Hibbs Alan R. (2004) Confocal Microscopy for Biologists. Kluwer Academic / Plenum Publishers, Page 187.

[17] I.I. Smalyukh, S.V. Shiyanovskii, D.J. Termine, O.D. Lavrentovich (2001) *Imaging&Microscopy,* Vol. 3, Pages 16 – 19.

[18] K J. Batenburg, and J. Sijbers (2009) *Optimal Threshold Selection for Tomogram Segmentation by Projection Distance Minimization*. IEEE Transactions on Medical Imaging, Vol. 28, Issue 5, Pages 676 – 686.

[19] P. Sankaran and V. K. Asari (2006) *Adaptive Thresholding Based Cell Segmentation for Cell-Destruction Activity Verification*. Applied Imagery and Pattern Recognition Workshop, IEEE

[20] Mancas M., Gosselin B., and Macq B. (2005) *Segmentation Using a Region Growing Thresholding*. Image Processing: Algorithms and Systems IV, San Jose, CA, USA, Vol. 5672, Pages 388 – 398.

[21] W. K. Pratt, (2007) *Digital Image Processing 4th Edition*, John Wiley & Sons, Inc., Los Altos, California, USA, Pages 562 – 564.

[22] N. Mustafa, N. A. Matisa, and M. Y. Mashor (2009) *Automated Multicells Segmentation of ThinPrep® Image Using Modified Seed Based Region Growing Algorithm*. Biomedical Soft Computing and Human Sciences, Vol. 14, Issue 2, Pages 41 – 47.

[23] X. Du and S. Dua. (2010) *Segmentation of Fluorescence Microscopy Cell Images Using Unsupervised Mining*. The Open Medical Informatics Journal, 2010, Vol. 4, Pages 41 – 49.

[24] M. Kass, A. Witkin, and D. Terzopoulos (1988) *Snakes: Active Contour Models*. International Journal of computer vision, Pages 321 – 331.

[25] A. A. Aly, S. B. Deris, and N. Zaki. (2011) *A Novel Image Segmentation Enhancement Technique Based On Active Contour And Topological Alignments.* Advanced Computing: An International Journal (ACIJ), Vol. 2, Issue 3.

[26] S. Osher, J. A. Sethian (1988), *Fronts propagating with curvature-dependent speed: Algorithms based on Hamilton-Jacobi formulations*, Journal of Computational Physics, Vol. 79, Pages 12 – 49.

[27] J. A. Sethian. (1999) *Level Set Methods and Fast Marching Methods: Evolving Interfaces in Computational Geometry, Fluid Mechanics, Computer Vision and Materials Science*. Cambridge University Press, Pages 30 – 31.

[28] J.A. Sethian. (1996) *Level Set Method: An Act of Violence - Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision and Materials Sciences*. American Scientist, Pages 7 – 9.

[29] Y. Lee. (2005) *Color Image Segmentation Using Multiphase Level Set Method with Adaptive Parameters*. Academic Dissertation, Institute of Computer Science and Information Engineering National Central University, Taiwan, Pages 11 – 15.

[30] O. Dzyubachyk, W. Niessen and E. Meijering. (2010) *Advanced Level-Set Based Multiple-Cell Segmentation And Tracking In Time-Lapse Fluorescence Microscopy Images*. Biomedical Imaging Group Rotterdam Erasmus MC, University Medical Center Rotterdam, Vol. 29 , Issue 3, Pages 852 – 867.

[31] G. Karypis, and V. Kumar (1998) *A Fast And High Quality Multilevel Scheme For Partitioning Irregular Graphs*. Society for Industrial and Applied Mathematics, Vol. 20, Issue 1, Pages 359 – 392.

[32] B. Hendrickson, and R. Leland. (1992) *An Improved Spectral Graph Partitioning Algorithm for Mapping Parallel Computations*. SIAM Journal on Scientific Computing, Society for Industrial and Applied Mathematics Philadelphia, PA, USA, Vol. 16, Issue 2, Pages 452 – 469.

[33] A. Pothen, H. D. Simon, and K.-P. Liou. (1990) *Partitioning sparse matrices with eigenvectors of graphs.* SIAM Journal on Scientific Computing, Society for Industrial and Applied Mathematics Philadelphia, PA, USA, Vol. 11, Issue 3, Pages 430 – 452.

[34] G. L. Miller, S.-H. Teng, W. Thurston, and S. A. Vavasis. (1993) *Automatic mesh partitioning*. Sparse Matrix Computations: Graph Theory Issues and Algorithms, Springer-Verlag, New York, Vol. 56, Pages 57 – 84.

[35] G. L. Miller, S.-H. Teng, and S. A. Vavasis. (1991) *A unified geometric approach to graph separators*. 32th Annual Symposium on Foundations of Computer Science, Pages 538 – 547.

[36] L. Hagen and A. Kahng. (1991) *Fast spectral methods for ratio cut partitioning and clustering*. International Conference on Computer Aided Design, IEEE, Pages 10 – 13.

[37] L. Hagen and A. Kahng. (1992) *A new approach to effective circuit clustering*. International Conference on Computer Aided Design, IEEE, Pages 422 – 427.

[38] B. Hendrickson and R. Leland. (1995). *A multilevel algorithm for partitioning graphs*. International Conference on Computer Aided Design, IEEE.

[39] Nakib, Amir, L. Najman, H. Talbot, and Patrick Siarry. (2011) *Graph partitioning*. ISTE – Wiley, Pages 251 – 274.

[40] Acharya, Tinku. (2005) *Image processing : principles and applications*. Hoboken, N.J, Pages 114 – 115.

[41] ImageJ Information and Documentation Portal: Subtract background. Available on (last accessing date: 20th, March, 2013): http://imagejdocu.tudor.lu/doku.php?id=gui:process:subtract_background

[42] Tony F. Chan, and Luminita A. Vese. (2001) *Active Contours Without Edges*. IEEE transactionson image processing, Vol. 10, Issue 2, Pages 266 – 277.

[43] Yan Zhang, Bogdan J. Matuszewski, Lik-Kwan Shark, and Christopher J. Moore. (2008) *Medical Image Segmentation Using New Hybrid Level-Set Method*. BioMedical Visualization, Fifth International Conference, Pages 71 – 76.

[44] A. Sarti, R. Malladi, and J. A. Sethian. (2000*) Subjective surfaces: a method for completing missing boundaries*. University of California, Vol. 97, Issue 12, Pages 6258 – 6263.

[45] A. Sarti, R. Malladi, and J. A. Sethian. (2002) *Subjective surfaces: a geometric model for boundary completion*. International Journal of Computer Vision, Vol. 46, Issue 3, Pages 201 – 221.

[46] A. Sarti, and G. Citti. (2001) *Subjective surfaces and Riemannian mean curvature flow of graphs*. Acta Math. Univ. Comenianae, Vol. 70, Issue 1, Pages 85 – 103.

[47] C. Zanella, M. Campana, B. Rizzi, C. Melani, G. Sanguinetti, P. Bourgine, K. Mikula, N. Peyrieras, and A. Sarti. (2010) *Cell segmentation From 3-D confocal images of early zebrafish embryogenesis*. IEEE Transactions on image processing, Vol. 19, Issue 3, Pages 770 – 781.

[48] J. Weichert, B. M. ter Haar Romeny, M. A. Viergever. (1998) *Efficient and Reliable Schemes for Nonlinear Diffusion Filtering*. IEEE Transactions on image processing, Vol. 7, Issue 3, Pages 398 – 410.

Appendix I

The derivation of Chan-Vese model energy function is shown in following steps.

In order to get the minimizing energy function, the integral function needs to be substituted by the energy function E.

$$E = \mu \cdot \delta\big(\phi(x,y)\big)\nabla|\phi(x,y)| + \mathrm{v} \cdot H\big(\phi(x,y)\big) + \lambda_1 \cdot |u_0(x,y) - c_1|^2 H(\phi(x,y)) + \lambda_2 \cdot |u_0(x,y) - c_2|^2 (1 - H(\phi(x,y)))$$

To simplify the calculation process, the energy function E is separate into four parts:

$E_1 = \mu \cdot \delta\big(\phi(x,y)\big)\nabla|\phi(x,y)|$, $E_2 = \mathrm{v} \cdot H\big(\phi(x,y)\big)$,

$E_3 = \lambda_1 \cdot |u_0(x,y) - c_1|^2 H(\phi(x,y))$, $E_4 = \lambda_2 \cdot |u_0(x,y) - c_2|^2 (1 - H(\phi(x,y)))$.

The minimizing energy function is obtained with the aid of Euler-Lagrange equation. For this function, the corresponding Euler-Lagrange function is:

$$E_\phi - \frac{\partial}{\partial x} E_{\phi x} - \frac{\partial}{\partial y} E_{\phi y} = 0$$

Firstly, to calculate E_ϕ, the four separated energy function can be calculated as:

$E_{\phi}^{1} = \mu \cdot \delta'\big(\phi(x,y)\big) \cdot |\nabla\phi(x,y)| + \mu \cdot \delta\big(\phi(x,y)\big) \cdot \frac{\partial}{\partial\phi}|\nabla\phi(x,y)| = \mu \cdot$

$\delta'\big(\phi(x,y)\big) \cdot |\nabla\phi(x,y)|,$

$E_{\phi}^{2} = v \cdot \delta(\phi(x,y)),$

$E_{\phi}^{3} = \lambda_1 \cdot |u_0(x,y) - c_1|^2 \delta(\phi(x,y)),$

$E_{\phi}^{4} = -\lambda_2 \cdot |u_0(x,y) - c_2|^2 \delta(\phi(x,y)),$

Then for th $\frac{\partial}{\partial x}E_{\phi x}$, the four results can be obtained as:

$$E_{\phi x}^{1} = \mu \cdot \delta_{\phi x}\big(\phi(x,y)\big) \cdot |\nabla\phi(x,y)| + \mu \cdot \delta\big(\phi(x,y)\big) \cdot \frac{\partial}{\partial\phi x}|\nabla\phi(x,y)|$$

$$= \mu \cdot \delta\big(\phi(x,y)\big) \cdot \frac{\partial}{\partial\phi x}\sqrt{{\phi_x}^2 + {\phi_y}^2}$$

$$= \mu \cdot \delta\big(\phi(x,y)\big) \cdot \frac{1}{2} \cdot \frac{1}{\sqrt{{\phi_x}^2 + {\phi_y}^2}} \cdot 2\phi_x$$

$$= \mu \cdot \delta\big(\phi(x,y)\big) \cdot \frac{\phi_x}{|\nabla\phi(x,y)|}$$

$\frac{\partial}{\partial x}E_{\phi x}^{1} = \mu \cdot \delta_{\phi}\big(\phi(x,y)\big) \cdot \phi_x \cdot \frac{\phi_x}{|\nabla\phi(x,y)|} + \mu \cdot \delta\big(\phi(x,y)\big) \cdot \frac{d}{dx}\left(\frac{\phi_x}{|\nabla\phi(x,y)|}\right),$

$\frac{\partial}{\partial x}E_{\phi x}^{2} = 0, \frac{\partial}{\partial x}E_{\phi x}^{3} = 0, \frac{\partial}{\partial x}E_{\phi x}^{4} = 0.$

Similarly, the four results of $\frac{\partial}{\partial y}E_{\phi y}$can be achieved in the same way:

$$E_{\phi y}^{1} = \mu \cdot \delta_{\phi y}\big(\phi(x,y)\big) \cdot |\nabla\phi(x,y)| + \mu \cdot \delta\big(\phi(x,y)\big) \cdot \frac{\partial}{\partial \phi y}|\nabla\phi(x,y)|$$

$$= \mu \cdot \delta\big(\phi(x,y)\big) \cdot \frac{\partial}{\partial \phi y}\sqrt{{\phi_x}^2 + {\phi_y}^2}$$

$$= \mu \cdot \delta\big(\phi(x,y)\big) \cdot \frac{1}{2} \cdot \frac{1}{\sqrt{{\phi_x}^2 + {\phi_y}^2}} \cdot 2\phi_y$$

$$= \mu \cdot \delta\big(\phi(x,y)\big) \cdot \frac{\phi_y}{|\nabla\phi(x,y)|}$$

$\frac{\partial}{\partial y}E_{\phi y}^{1} = \mu \cdot \delta_{\phi}\big(\phi(x,y)\big) \cdot \phi_y \cdot \frac{\phi_y}{|\nabla\phi(x,y)|} + \mu \cdot \delta\big(\phi(x,y)\big) \cdot \frac{d}{dy}\left(\frac{\phi_y}{|\nabla\phi(x,y)|}\right)$,

$\frac{\partial}{\partial y}E_{\phi y}^{2} = 0, \frac{\partial}{\partial y}E_{\phi y}^{3} = 0, \frac{\partial}{\partial y}E_{\phi y}^{4} = 0.$

Therefore, combining all the separated results together in the Euler-Lagrange equation, it can be expressed as:

$$E_\phi - \frac{\partial}{\partial x}E_{\phi x} - \frac{\partial}{\partial y}E_{\phi y}$$

$$= \mu \cdot \delta_\phi\big(\phi(x,y)\big) \cdot |\nabla\phi(x,y)| + v \cdot \delta\big(\phi(x,y)\big) + \lambda_1 \cdot |u_0(x,y)$$

$$- c_1|^2 \delta\big(\phi(x,y)\big) - \lambda_2 \cdot |u_0(x,y) - c_2|^2 \delta\big(\phi(x,y)\big) - \mu$$

$$\cdot \delta_\phi\big(\phi(x,y)\big) \cdot \phi_x \cdot \frac{\phi_x}{|\nabla\phi(x,y)|} - \mu \cdot \delta\big(\phi(x,y)\big) \cdot \frac{d}{dx}\left(\frac{\phi_x}{|\nabla\phi(x,y)|}\right)$$

$$- \mu \cdot \delta_\phi\big(\phi(x,y)\big) \cdot \phi_y \cdot \frac{\phi_y}{|\nabla\phi(x,y)|} - \mu \cdot \delta\big(\phi(x,y)\big)$$

$$\cdot \frac{d}{dy}\left(\frac{\phi_y}{|\nabla\phi(x,y)|}\right)$$

$$= \mu \cdot \delta_\phi\big(\phi(x,y)\big) \cdot |\nabla\phi(x,y)| + v \cdot \delta\big(\phi(x,y)\big) + \lambda_1 \cdot |u_0(x,y)$$

$$- c_1|^2 \delta\big(\phi(x,y)\big) - \lambda_2 \cdot |u_0(x,y) - c_2|^2 \delta\big(\phi(x,y)\big) - \mu$$

$$\cdot \delta_\phi\big(\phi(x,y)\big) \cdot |\nabla\phi(x,y)| - \mu \cdot \delta\big(\phi(x,y)\big) \cdot \nabla\left(\frac{\nabla\phi(x,y)}{|\nabla\phi(x,y)|}\right)$$

$$= \delta\big(\phi(x,y)\big)$$

$$\cdot \Bigg(v + \lambda_1 \cdot |u_0(x,y) - c_1|^2 - \lambda_2 \cdot |u_0(x,y) - c_2|^2 - \mu$$

$$\cdot \nabla\left(\frac{\nabla\phi(x,y)}{|\nabla\phi(x,y)|}\right)\Bigg) = 0$$

Therefore, this gives:

$$\delta\big(\phi(x,y)\big) \cdot \Big(\mu \cdot \nabla\left(\tfrac{\nabla\phi(x,y)}{|\nabla\phi(x,y)|}\right) - v - \lambda_1 \cdot |u_0(x,y) - c_1|^2 + \lambda_2 \cdot |u_0(x,y) -$$

$$c_2|^2\Big) = 0,$$

Appendix II

The derivation of hybrid method formula is shown in the following steps.

$$F(\phi) = -\alpha \int_{\Omega} (I - \mu) H(\phi) d\Omega + \beta \int_{\Omega} g|\nabla H(\phi)| d\Omega$$

To simplify the calculation process, function F is separate into two parts:

$F_1 = -\alpha(I - \mu)H(\phi)$, $F_2 = \beta g|\nabla H(\phi)| = \beta g \delta(\phi)|\nabla \phi|$

The minimizing energy function is obtained with the aid of Euler-Lagrange equation. For this function, the corresponding Euler-Lagrange function is:

$$F_\phi - \frac{\partial}{\partial x} F_{\phi x} - \frac{\partial}{\partial y} F_{\phi y} = 0$$

Firstly, to calculate F_ϕ, its two parts can be calculated as:

$F_\phi^1 = -\alpha(I - \mu)H_\phi(\phi) = -\alpha(I - \mu)\delta(\phi)$,

$F_\phi^2 = \beta g \delta_\phi(\phi)|\nabla \phi| + \beta g \delta(\phi) \frac{\partial}{\partial \phi}|\nabla \phi| = \beta g \delta_\phi(\phi)|\nabla \phi|$.

Then for $\frac{\partial}{\partial x} F_{\phi x}$, the four results can be obtained as:

$\frac{\partial}{\partial x} F_{\phi x}^1 = 0$,

Since

$F_{\phi x}^2 = \beta g \delta_{\phi_x}(\phi)|\nabla\phi| + \beta g \delta(\phi)\frac{\partial}{\partial \phi_x}|\nabla\phi| = \beta g \delta(\phi) \cdot \frac{1}{2} \cdot \frac{1}{|\nabla\phi|} \cdot 2\phi_x = \beta g \delta(\phi) \cdot \frac{\phi_x}{|\nabla\phi|}$,

Therefore, $\frac{\partial}{\partial x}F_{\phi x}^2 = \beta g \delta_{\phi}(\phi) \cdot \frac{\phi_x^2}{|\nabla\phi|} + \beta g \delta(\phi) \cdot \frac{d}{dx}\left(\frac{\phi_x}{|\nabla\phi|}\right)$.

Similarly, the results of $\frac{\partial}{\partial y}F_{\phi y}$can be achieved in the same way:

$\frac{\partial}{\partial y}F_{\phi y}^1 = 0, \frac{\partial}{\partial y}F_{\phi y}^2 = \beta g \delta_{\phi}(\phi) \cdot \frac{\phi_y^2}{|\nabla\phi|} + \beta g \delta(\phi) \cdot \frac{d}{dy}\left(\frac{\phi_y}{|\nabla\phi|}\right)$

Therefore, combining all the separated results together in the Euler-Lagrange equation, it can be expressed as:

$$F_{\phi} - \frac{\partial}{\partial x}F_{\phi x} - \frac{\partial}{\partial y}F_{\phi y} = -\alpha(I-\mu)\delta(\phi) - \beta g \delta(\phi) \cdot \nabla\left(\frac{\phi_x}{|\nabla\phi|}\right)$$

$$= -\delta(\phi)\left(\alpha(I-\mu) + \beta g \nabla\left(\frac{\phi_x}{|\nabla\phi|}\right)\right) = 0$$

Appendix III

As shown in the flowchart below, the Chan-Vese algorithm contains the following five steps:

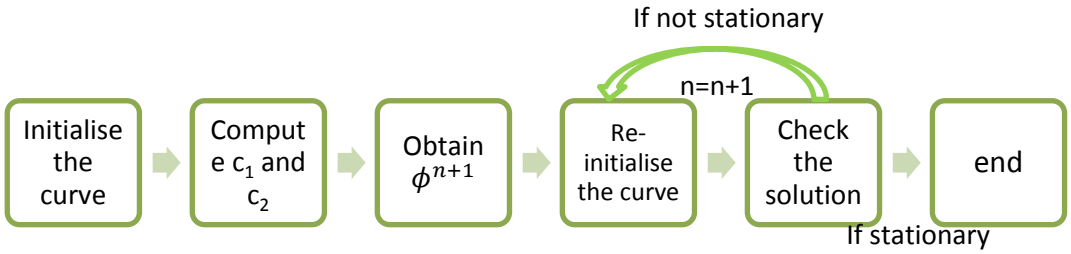
Step 1: Initialise ϕ^0 by $\phi_0, n = 0$.

Step 2: Compute c_1 and c_2 with $c_1(\phi) = \frac{\int_\Omega u_0(x,y)H(\phi(x,y))dxdy}{\int_\Omega H(\phi(x,y))dxdy}$ and $c_1(\phi) = \frac{\int_\Omega u_0(x,y)\left(1-H(\phi(x,y))\right)dxdy}{\int_\Omega\left(1-H(\phi(x,y))\right)dxdy}$.

Step 3: Solve the PDE in ϕ from $\delta(\phi(x,y)) \cdot \left(\mu \cdot \nabla\left(\frac{\nabla\phi(x,y)}{|\nabla\phi(x,y)|}\right) - v - \lambda_1 \cdot |u_0(x,y) - c_1|^2 + \lambda_2 \cdot |u_0(x,y) - c_2|^2\right) = 0$ to obtain ϕ^{n+1}.

Step 4: Re-initialise ϕ locally to the signed distance function to the curve.

Step 5: Check whether the solution is stationary. If not, $n = n + 1$ and repeat.



Flow chart of Chan-Vese model implementation procedures

Hybrid method procedure is similar to Chan-Vese model but consists of seven steps and concluded in the following flow chart.

Step 1: Initialise ϕ^0 by $\phi_0, n = 0$.

Step 2: Solve the PDE in ϕ using $\phi_t = \alpha(I - \mu) + \beta div(g\nabla\phi)$, to obtain ϕ^k where α, β are parameters that can be set by users.

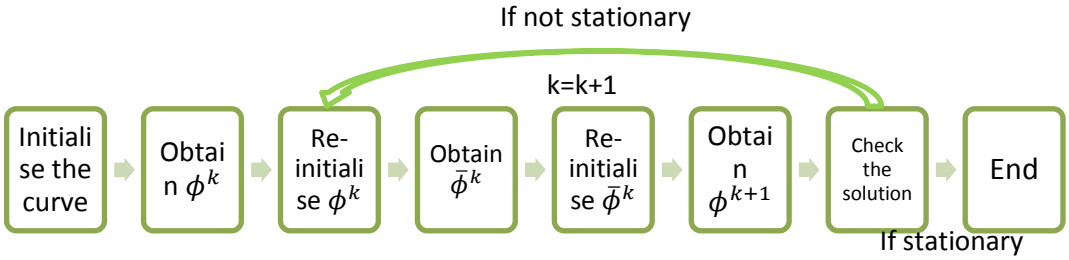
Step 3: Re-initialise ϕ^k. Re-initialisation of an embedding function ϕ is the process to make $|\phi| = 1$ while remaining the embedded curve (the zero set) unchanged.

Step 4: Update ϕ^k to obtain $\bar{\phi}^k$ using $\bar{\phi}^k = \phi^k + \Delta_t\alpha(I - \mu)$ with the predefined time step Δ_t.

Step 5: Re-initialise $\bar{\phi}^k$.

Step 6: Update $\bar{\phi}^k$ to obtain ϕ^{k+1} via solving the PDE $\phi_t = \beta div(g\nabla\phi)$ using AOS scheme.

Step 7: Check whether the solution is stationary. If not, $n = n + 1$ and repeat.



Flow chart of hybrid method implementation procedures