Bayesian Model Based Tracking with Application to Cell Segmentation and Tracking

by

Nezamoddin Nezamoddini-Kachouie

A Thesis
presented to the University of Waterloo
in fulfillment of the
thesis requirement for the degree of
Doctor of Philosophy
in
Systems Design Engineering

Waterloo, Ontario, Canada 2008

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I understand that my thesis may be made electronically available to the public.

Nezamoddin Nezamoddini-Kachouie

Abstract

The goal of this research is to develop a model-based tracking framework with biomedical imaging applications. This is an interdisciplinary area of research with interests in machine vision, image processing, and biology.

This thesis presents methods of image modeling, tracking, and data association applied to problems in multi-cellular image analysis, especially hematopoietic stem cell (HSC) images at the current stage. The focus of this research is on the development of a robust image analysis interface capable of detecting, locating, and tracking individual hematopoietic stem cells (HSCs), which proliferate and differentiate to different blood cell types continuously during their lifetime, and are of substantial interest in gene therapy, cancer, and stem-cell research.

Such a system can be potentially employed in the future to track different groups of HSCs extracted from bone marrow and recognize the best candidates based on some biomedical-biological criteria. Selected candidates can further be used for bone marrow transplantation (BMT) which is a medical procedure for the treatment of various incurable diseases such as leukemia, lymphomas, aplastic anemia, immune deficiency disorders, multiple myeloma and some solid tumors.

Tracking HSCs over time is a localization-based tracking problem which is one of the most challenging tracking problems to be solved. The proposed cell tracking system consists of three inter-related stages:

- Cell detection/localization,
- The association of detected cells,
- Background estimation/subtraction.

that will be discussed in detail.

Acknowledgments

I would like to take this opportunity to thank everybody who has contributed to the PhD work. Very sincere thanks go to my supervisor Prof. Paul Fieguth from the Department of Systems Design Engineering at the University of Waterloo. I would not have been able to carry out the PhD without his guidelines and constant support. He always keeps his door open in order to discuss the research with passion and great enthusiasm. I will always remain very grateful to him for everything that I learned from him and I am glad to have him as my advisor throughout the journey of the doctoral work.

I would also like to thank Prof. Eric Jervis and John Ramunas from the Chemical Engineering Department of the University of Waterloo for providing microscopic image sequences for this research and valuable inputs on biological properties of HSCs.

My sincere thanks go to Prof. Edward Jernigan, a dedicated and exemplary professor and former chair of the Department of Systems Design Engineering, for his continuous support.

I would like to thank my wife Sholeh for her love, support, and always believing in me through my long, and maybe unfinished journey of science and education. I would like to thank my mom for the continuous support of my dreams.

This research was supported in various ways by grants and/or scholarships from the National Sciences and Engineering Council, Ontario Graduate Scholarships, the Faculty of Engineering and the Department of Systems Design Engineering. I am truly appreciative of this aid.

My sincere thanks go to the members of Vision and Image Processing Lab for their inspiration and social support. Their support was very important throughout the entire journey of the doctoral work.

To Sholeh and Kian

Contents

List of Tables

List of Figures

Chapter 1

Introduction

The automatic acquisition of huge numbers of digital images has been made possible by advances in and the low cost of digital imaging. Several new applications have been introduced relying on the acquisition and analysis of long streams of images. In most video analysis applications, the goal is to track one or multiple moving objects over the data stream. Tracking has a broad range of applications including some well-known applications such as air traffic control [115], robot control [2, 111], ocean surveillance [67], and some recent applications such as automated vehicle control [107] and, of particular interest to this research, cell tracking [26, 103]. However, manual tracking methods are so onerous that automated tracking methods are mandatory for handling such huge amount of videos.

1.1 Context and Motivation

Drug discovery relies partly on genomics, the study of genes, to identify drug targets. Understanding the dynamic function and interactions of proteins as potential drug targets is required to discover drug-cell interactions. Therefore, a better understanding of cell

behavior is very important in drug and disease research. Cell size, shape, and motility may play a key role in stem-cell specialization or cancer development. Hence, observing the specimen cells and measuring their geometric parameters helps pathologists to find out how cancerous cells infect a cellular organism and to make diagnostic decisions. However manual methods of inferring these values are tedious and difficult due to corrupted and blurred images, the presence of clutter objects, fixing eyes for a long time and repeating the same task for several different cell types. Furthermore, with the extent of cell imaging data ever increasing, manual tracking becomes progressively impractical. Due to these problems and the multitude of applications which rely on the results of cell segmentation, recognition, and tracking, there is an immense need for automated systems [8, 17, 33, 37, 54, 90, 96, 128].

As a result, an automatic or semi-automatic cell image analysis system has a crucial importance in the study of cell behaviour and, in turn, in drug and disease research. Automated processing of cellular data to extract cell features such as cell size, shape, and motility is an interesting yet challenging field in bioinformatics. Such an automated system relies on the collection, processing, and analysis of huge amounts of data including long streams of cellular image sequences.

An automated tracking system will require a computer to perform automatic object tracking, usually under challenging conditions, which also presents a very attractive yet difficult research problem for researchers in computer vision and digital image processing.

1.2 Problem Description

There are many applications of cell segmentation/localization, detection/recognition, and tracking in microscopy image sequences. Depending on the nature of the tracking problem it may belong to one of the following categories:

1. Segmentation-based tracking: objects/regions have well recognizable boundaries or regions such that they can be well segmented.

    - This case turns to the third category, below, if segmented objects/regions can be discriminated based on their features.
    - This case turns to the fourth category, below, if segmented objects/regions can not be discriminated based on the extracted features.

2. Detection-based tracking: objects have poor contrast such that they can not be well segmented, however the presence of objects can be detected.

    - This case turns to the third category, below, if detected objects can be discriminated based on their features.
    - This case turns to the fourth category, below, if detected objects can not be discriminated based on the extracted features.

3. Recognition-based tracking: objects can be discriminated and recognized based on their features. In this case, object recognition plays the major role to solve the tracking problem. Each recognized object in the present time may either be the extension of an existing track in the previous time step or be a new track.

4. Localization-based tracking: objects can be segmented and/or detected, however they can not be recognized. In this case objects/regions have the same visual appearance, and/or features such that they can not be discriminated based on their features.

Based on the tracking application, therefore, segmentation, detection, recognition, and localization are crucial stages.

1.3 Thesis Goals and Contributions

The goal of this research is to develop a model-based tracking framework with biomedical imaging applications. This is an interdisciplinary area of research with interests in machine vision, image processing, and biology. We have the opportunity to access real data which is provided by our collaborators in the Department of Chemical Engineering at the University of Waterloo using advanced microscope systems.

Due to the large number of cell types with different features and behaviours, designing a universal cell tracking system is impractical. In this research, we focus on Hematopoietic Stem Cells (HSCs), which proliferate and differentiate to different blood cell types continuously during their lifetime, and are of substantial interest in gene therapy, cancer, and stem-cell research. A typical phase contrast HSCs image is depicted in Fig. 1.1. Tracking HSCs over time is a problem that belongs to the fourth tracking category, localization-based tracking, which is one of the most challenging tracking problems to be solved. The key challenges of our tracking problem can be summarized as:

1. To keep cells alive and healthy, light exposure must be controlled during their life cycle to minimize phototoxicity. Therefore it is desired to limit light exposure in each frame and to sample the frames as far apart as possible, leading to infrequent, poorly-contrasted images, directly at odds with the data desired for easy tracking: frequent, high-contrast images. Cell staining techniques may be used to increase the contrast between cell and background, however different parts of tissue are undesirably stained unevenly, causing inhomogeneity.

2. The blood stem cells which are of interest here in our research, all have the same visual appearance and cannot be discriminated visually.

3. To track a particular cell over time, the association task becomes crucial.

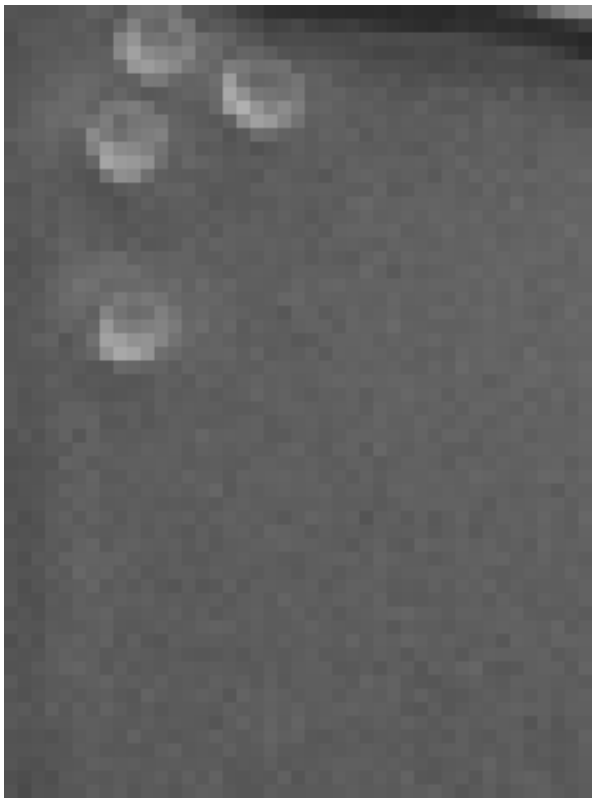


Figure 1.1: A typical HSCs phase contrast image.

In this research, a model-based cell tracker is designed to locate and track individual cells. The proposed cell tracking system consists of three inter-related stages:

- Cell detection/localization (Chapter 6),
- The association of detected cells (Chapter 8),
- Background estimation/subtraction (Chapter 7).

and will be explained in detail in the rest of this thesis.

1.4 Thesis Outline

The outline of this research is summarized as follows.

- *Stem Cell Studies - Living Cell Imaging (Chapter 2)*
- *Mathematical Background (Chapter 3)*
- *The Proposed Approach (Chapter 4)*
- *Image Denoising (Chapter 5)*
- *Cell Segmentation and Localization (Chapter 6)*
- *Background Estimation (Chapter 7)*
- *Cell Tracking (Chapter 8)*
- *Conclusions and Future Work (Chapter 9)*

Chapter 2

Stem Cell Studies - Living Cell Imaging

The rapid growth of bioinformatics and biotechnology research relies on computer hardware and software technologies to collect, process and analyze ever-increasing amounts of data. More closely related to the community of image and video researchers, digital cytometry [37] has been recently introduced to adapt and extend image processing techniques to analyze and extract cell properties from microscopic cell images. By applying advanced techniques in digital image processing and pattern recognition to a huge number of bio-cellular images, digital cytometry can improve our understanding of cellular and inter-cellular events so that significant progress and new discoveries in biological and medical research may be achieved. In this chapter an overview of cytometry and cell microscopic imaging are presented.

2.1 Stem Cells

An embryo grows up to be a fetus, a fetus grows up to be a baby, a baby grows up to be a child, and a child grows up to be an adult. A continuous renewal process develops, maintains, and repairs the human body through its life. The building blocks of this process are stem cells. Stem cells proliferate and differentiate to all different cell types, including bone cells, muscle cells, blood cells, skin cells, and several other cell types. Stem cells continue the development, maintenance, and repair of human body by proliferation and differentiation to many different cell types. In this section we have a brief overview of the stem cell types that we are interested in and we have used in our research.

2.1.1 Embryonic Stem Cells

When an egg and sperm fuse to form an embryo, those early cells will eventually give rise to every type of cell in the adult human body. By the stage at which the embryo would implant in its mothers womb it has become a hollow ball of cells. The inner cells of this ball are what is known as pluripotent, because they will eventually form most, but not all of the cells of the baby. These pluripotent stem cells get ever more specialized as the embryo develops, forming for example blood stem cells which give rise to blood cells and skin stem cells which give rise to skin cells. Embryonic stem cells (ESCs) have a remarkable capacity to proliferate and differentiate to other cell types. They divide and differentiate to the specialized cells of the body which in turn make up the various tissues and organs. Their ability to differentiate to desirable phenotypes has motivated clinical interests. However, progress in the analysis of ESC functional properties is required for development of clinically applicable procedures for stem cell transplantation and treatment of various incurable diseases. ESCs can reverse Parkinson's, save defective hearts, and

Figure 2.1: A Phase Contrast Image of a cluster of 10 NSCs with irregular shapes.

repair heart muscle. Recent research has proven that ESCs can insulate bare nerve fibers in an animal model [96]. A new technique which has successfully been performed in mice ESCs turns the ESCs into the myelin building blocks of tissue. Hence, they can hopefully be used to treat diseases like multiple sclerosis or spinal cord injuries.

Due to the universal attributes of ESCs, there has been great interest to develop a practical automated approach to measure and extract ESC properties from microscopic cell images and track individual cells over time. To accomplish this task and locate the ESC clusters over time in a sequence of frames, and in turn to perform ESC cluster motion analysis, first ESC clusters must be segmented.

2.1.2 Neural Stem Cells

Neurons do not have the capability of self-renewal or differentiation to other cell types, hence after injuries in the central nervous system they cannot regenerate the damaged nervous system. In contrast neural stem cells (NSCs) as building blocks of the brain can proliferate and differentiate into all neural phenotypes. Neural stem cells can change to the motor neurons which in turn these motor neurons differentiate into spinal motor neuron cells which transmits messages from the brain to the spinal cord in the human body. Therefore they are an ideal cell type which can be used as potential transplantable tissue to repair damaged neurodegenerative processes, such as Alzheimer's and to repair brain injuries such as stroke.

There is great interest to discover the cellular and molecular mechanisms which control NSC proliferation and differentiation. We should also point out that from an image processing point of view NSCs and ESCs are quite similar and belong to the same category. However the NSC microscope images are noisier and have less recognizable cell boundaries, hence segmentation and tracking of NSC data are more difficult than of ESC. Typical clusters of NSCs are depicted in Figs. 2.1 and 2.2.

2.1.3 Hematopoietic Stem Cells

Pluripotent hematopoietic stem cells (HSCs) proliferate and differentiate to replace mature blood cells which are at the end of their life-span. HSCs can differentiate to white blood cells, red blood cells and platelets. There is a small population of self renewal HSC in the bone marrow of healthy adults which can differentiate into mature blood cells of all hematopoietic lineages. A group of 4 HSCs is depicted in Fig. 2.3. Clinical procedures for stem cell transplantation contain the following steps:

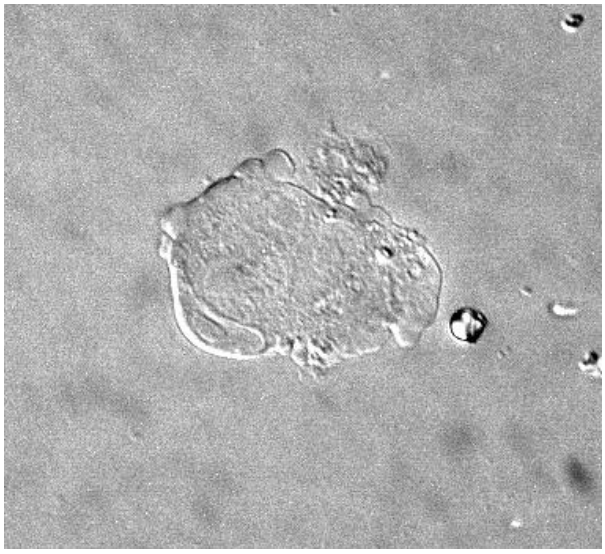


Figure 2.2: A Phase Contrast Image showing a cluster of NSCs with irregular shapes.

- Applicable HSC isolation procedure
- Analysis of HSC functional properties
- Experimental manipulating approaches of HSC in culture

Based on recent research [84, 125] HSC can even contribute to non-hematopoietic tissues such as liver, skin and muscle. Such contributions can be used in HSC medicine regeneration, such that after transplantation, HSCs are able to reconstruct recipient's tissue. Stem cell biology is studied using experimental models and have been proven by mouse transplantation assays. These models have been used to study different characterization of stem cells such as immuneophenotype, homing ability and kinetics.

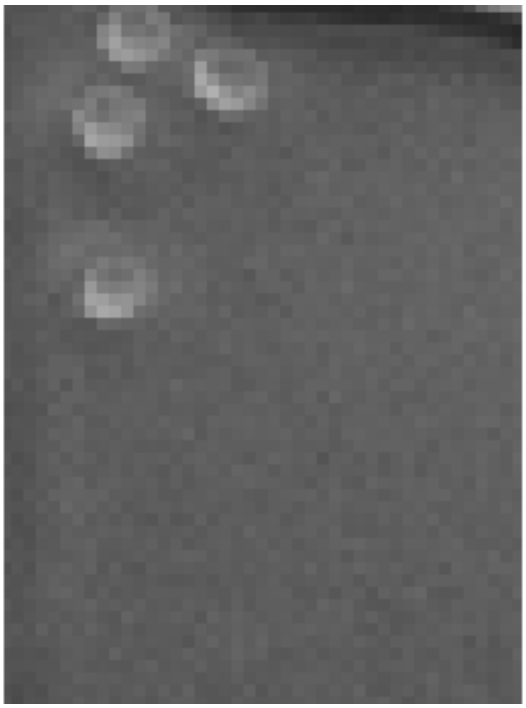


Figure 2.3: A HSC Phase Contrast Image showing a group of 4 HSCs with uniform shapes.

HSC Renewal *in vivo* and *in vitro*

Continuous activation of a small population of HSCs supports the reproduction of blood cells during human life. HSCs are able to split to maintain this small population. At least half of the split daughter cells have the same proliferation potential as the original parent HSCs [28]. Self renewal of HSCs in vivo has been proven and remarkable heterogeneity has been seen for individual evaluated HSCs in their self renewal responses. There is not much known about the regulation of HSC self renewal properties in molecular level. To study

how different growth factors effects loss or preservation of HSCs in vitro, HSC proliferation rate and the increase in the population must be measured by observation and tracking of HSCs over time.

2.2 Cytometry

The study of the functional relationship between cytomes, cells, and metabolic pathways (proteomic) through genetic control (genomic) is known as *cytometry*. This field studies the cell characteristics, measures cell attributes and extracts cell features. Extracted features may belong to physical features such as shape and area or biological features such as types of protein contents. Analytical biology uses these features for analytical diagnosis. A practical approach to measure cell properties is *image cytometry*. This area studies the measurement and extraction of cell properties from microscopic cell images. Spectral, spatial and temporal analyzes of cell images can provide inter and intra cellular information about a single cell or a group of cells. Because of the huge amounts of data, the required accuracy in measurements and the reproducibility of the results, manual interpretation is a very tedious task and sometimes impossible. These attributes of the problem call for computerized methods to analyze cell images.

2.3 HSC Sample Preparation and Digital Microscopy Imaging

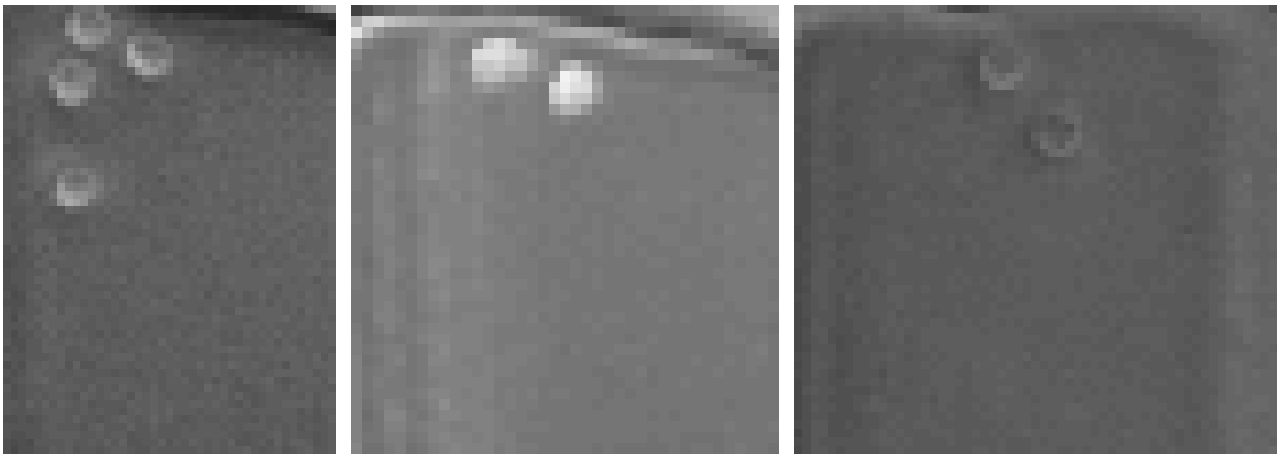
HSC samples must be extracted and processed before imaging. To prepare HSC samples, mice as bone marrow donors are bred and maintained in the animal facility at the BC cancer research center in micro isolators and are fed with sterilized food and water.

2.3.1 HSC Sample Preparation

HSC sample preparation is a two stage process:

1. Extract and process the bone marrow from the mouse to prepare HSCs. HSCs are located in the adult mouse bone marrow. The HSC content of specific subsets of SP cells are measured to analyze the properties of different fractions of SP cells. To track HSCs and investigate how different growth factors affect the initial rate of HSC proliferation in vitro, a highly HSC-enriched fraction is used. As described in [33], bone marrow cells are isolated from sacrificed mice and are enriched for stem cells. Briefly, the cells are incubated with certain fluorescent dyes and dye-conjugated antibodies that are useful for distinguishing between hematopoietic stem cells and non-stem cells, specifically Hoechst dye, Rhodamine dye and Lin which is a dye conjugated antibody. Then the stem cells are separated from the other cells based on their fluorescence using a fluorescence activated cell sorter. This sorter measures the fluorescence of each cell at various wavelengths and directs the cell into the appropriate container based on its fluorescence profile.

2. Process and culture the HSCs. Approximately 30 HS cells were loaded in 1 ml of standard HSC medium (alpha-MEM with 7.5% newborn calf serum and 2.5% fetal bovine serum) into a 5.8 micron gap between a cover-slip and a 1.3 mm x 10 mm rectangle from a 1 mm-thick glass slide. The culture chamber had been precoated with 1% BSA solution by incubating for 4 hours at 37° C. The culture chamber was placed in an enclosure containing humidified air with 5% CO2 at 37° C on a Zeiss Axiovert 200 (Carl Zeiss, Inc., Thornwood, NY) inverted microscope.



(a) (b) (c)

Figure 2.4: HSC Phase Contrast Microscopic Images: (a) Phenotype 1. (b) Phenotype 2 (c) Phenotype 3

2.3.2 Digital Microscopy Imaging

Digital cytometry couples digital cell images, digital image processing techniques and biology knowledge to measure the cell properties. To produce digital cell images, a digital camera is combined with a microscope. This digital microscope can directly record cell related biological events over time during an adjustable time interval. The time interval is adjusted based on cell types and their life time and is also restricted to a technical lower limit based on the digital microscope. There are different digital microscopic imaging techniques including fluorescence, phase contrast, differential interference contrast, bright-field and modulation contrast microscopy.

The interaction of light and a specimen generates image contrast. This interaction can be identified through a variety of mechanisms including reflection, diffraction, refraction, absorption, fluorescence and polarization. The image contrast can be improved by either

modification of optical components of microscope or photographic manipulation. In this section two of the optical microscopy techniques which have been developed to enhance image contrast are reviewed.

Phase Contrast (PC) Imaging

A typical HSC microscopic image is depicted in Fig. 2.4. Most of the living biological specimens look transparent if we observe them by a digital optical microscope under brightfield illumination. To have a better contrast and improve visibility, usually observers reduce the microscope diaphragm's opening size which causes some diffraction artifacts and resolution loss. PC microscopy was introduced by Dutch physicist Frits Zernike in 1934 to improve the contrast without resolution loss. This method was employed at first in early 1930's for the test of telescope mirrors and after a long time was considered as an improvement method in microscopes to observe transparent biological specimens. PC has been used as an efficient technique to increase the contrast of unstained biological specimens. This method is used to observe inter and intra living cell events over time without significant loss in resolution [1]. In the phase contrast technique variations in phase are translated to corresponding changes in amplitude by an optical mechanism such that these variations can be visualized as differences in image contrast. The main advantage of this technique is observing living cells without staining or fixing them and recording minute details of their dynamics in their natural state with high contrast and sharp clarity.

Differential Interference Contrast (DIC) Microscopy

Another important microscopic imaging modality is DIC. It has been used widely, not only for motion analysis of living cells, but also for the study of biological structure of specimens. In contrast with fluorescence imaging, which is limited to photo-dying, DIC can

be used for the imaging of transparent specimens which are not visible in usual transmitted light microscopy imaging. This is very important, especially in cases where specimen staining is not possible and staining would kill the specimen. Two mutually coherent waves, which are known as shear with a very small differential displacement, are phase shifted relative to each other. The variations in the refractive index across the living cell cause a phase difference between the shears. The gradient of high and low spatial frequencies present in the living cell is displayed effectively by this method which produces a high contrast image. Bright regions in this image are caused by increasing of the optical paths along a reference direction, while decreasing the optical path along the reference direction inverses the contrast. As a result the larger gradient of the optical path difference increases the image contrast. To keep cells alive and dynamically active, light exposure must be controlled during their life cycle. The limited light exposure and cell transparency both contribute to the very low contrast of typical microscopic cell images.

Chapter 3

Mathematical Background

A tracking system basically might address object localization/segmentation and tracking. In cases of tracking objects in noisy image sequences, the tracking system might also address preprocessing tasks such as denoising and image enhancement. An overview of the mathematical background of our research is presented in this chapter as follows:

- *Image Denoising*
- *Image Segmentation and Localization*
- *Background Estimation*
- *Tracking*

3.1 Denoising

The goal of denoising is to remove the noise and retain important signal features as much as possible. To achieve this goal, different approaches use either linear or nonlinear filters for denoising. Although Wiener filter [57], a typical linear filter, achieves the optimal

denoising for stationary signals, denoising using stationary linear filters does not perform effectively for non-stationary signals and functions with discontinuities.

As a result to denoise non-stationary signals corrupted by additive noise, stationary linear filters, which consist of convolving the image with a constant matrix to obtain a linear combination of neighborhood values, can produce a blurred and smoothed image with poor feature localization and incomplete noise suppression.

To overcome these shortcomings, local linear filters such as Lee filter which is a linear approximation filter based on the minimum mean-square error (MMSE) [24] and nonlinear filters have been proposed for denoising non-stationary signals. Much research has recently focused on signal denoising using nonlinear techniques in the setting of additive white Gaussian noise. One of the most common nonlinear denoising techniques is wavelet-based denoising [41, 88]. Signal approximation using wavelet decomposition followed by wavelet coefficient thresholding allows an effective representation of signal discontinuities. In this way decomposed wavelet coefficients will be thresholded to separate the signal and noise; as a result wavelet denoising can be used to remove the noise while preserving the signal characteristics. Researchers have employed two different approaches to nonlinear wavelet-based denoising: first, known as wavelet thresholding [40, 41, 121], a hard threshold function keeps a coefficient if it is larger than the threshold and sets it to zero otherwise; second, wavelet shrinkage with a soft thresholding function takes the wavelet coefficient and shrinks it toward zero by the threshold. Both approaches are nonlinear and operate on one wavelet coefficient at a time.

3.1.1 Wavelet Image Denoising

Wavelet denoising [40, 41, 121] is used to recover the original signal from the noisy one by removing the noise. In contrast with denoising methods that simply smooth the signal

by preserving the low frequency content and removing the high frequency components, the frequency contents and characteristics of the signal would be preserved during wavelet denoising which includes three steps:

- Signal decomposition by wavelet transform.
- Noise removal by thresholding.
- Signal reconstructions by inverse wavelet transform.

The first and third steps are linear while the thresholding step is a nonlinear process and is the main attribute of wavelet denosing which singles it out from linear denoising techniques.

Denoising Concept

To explain the wavelet denoising procedure, assume S to be the original M by M image where i and $j = 1, 2, \ldots, M$ and it is corrupted with additive noise n:

$$I[i, j] = S[i, j] + n[i, j] \tag{3.1}$$

n is identically distributed and independent of S. The goal of denoising is to estimate $\hat{S}[i, j]$ of $I[i, j]$ by removing the noise $n[i, j]$. In the first step of wavelet denoising the observed image I is transformed into the wavelet domain. Then, the wavelet coefficients are thresholded and transformed back to reconstruct the image. Let W_D and W_R be the forward wavelet decomposition and inverse wavelet reconstruction, respectively. Let τ and T be threshold value and thresold operator respectively. So the wavelet thresholding can be summarized as

$$\begin{aligned} I_w &= W_D(I) \\ I_\tau &= T(I_w, \tau) \\ \hat{S} &= W_R(I_\tau) \end{aligned} \tag{3.2}$$

Wavelet denoising reconstructs an estimated image with less noise by setting the small wavelet coefficients (dominated by noise), which are less than a specific threshold τ, to zero. In brief we can say wavelet thresholding performs well and is an efficient denoising approach.

3.1.2 Hard Thresholding vs. Soft Thresholding

Wavelet denoising methods are divided into two major groups: hard and soft thresholding [41, 121].

Hard Thresholding

Let T_H be the hard thresolding operator, so that

$$\begin{aligned} T_H(I_w, \tau) &= I_w \quad \text{if} \quad |I_w| > \tau \\ &= 0 \quad \text{otherwise} \end{aligned} \tag{3.3}$$

which keeps the wavelet coefficient if it is larger than the threshold τ and sets it to zero otherwise.

Soft Thresholding (Shrinkage)

In contrast, the soft thresholding T_S is:

$$T_S(I_w, \tau) = sign(I_w) \ \cdot \ max(0, |I_w| - \tau) \tag{3.4}$$

shrinks the wavelet coefficient by the threshold τ toward zero. Although hard thresholding mathematically seems more natural, soft thresholding makes thresholding smoother and has the following advantages:

- The noise coefficients, which may be passed by hard thresholding and appear as unpleasant artifacts in the output image, are shrunk by soft thresholding.
- When the noise energy is high, the denoised image produced by soft thresholding is visually more pleasant than that with hard thresholding. Generated artifacts, due to the discontinuity of hard thresholding, degrade the recovered image.

VisuShrink Image Denoising Technique

An important issue when using wavelet thresholding for image denoising is how to determine the threshold τ. Though selecting a small threshold may produce an output image close to the input, recovered image may still be noisy. On the other hand, choosing a large threshold may yield a smoothed image by setting most of the wavelet coefficients to zero and will lose desirable image details. The VisuShrink denoising technique proposed by Donoho and Johnstone [40, 41] uses the universal threshold given by

$$\tau_u = \sigma_n \sqrt{2log(M)} \tag{3.5}$$

Where σ_n and M are the noise variance and the number of image pixels respectively. Donoho and Jonstone have proven that the maximum of any M independent and identically

distributed (iid) values with high probability is less than the universal threshold τ_u. As M is increased the probability will be closer to one, so with a high probability pure noise signals are set to zero. The universal threshold is obtained by considering the constraint that the noise is less than the threshold with high probability as M increases, hence it tends to be high for large values of M. As a result it will shrink many noisy wavelet coefficients to zero and produce smoothed estimated image.

BayesShrink Image Denoising Technique

The subband Wavelet coefficients I_w of a natural image can be described by the Generalized Gaussian Distribution (GGD) [22, 23] as:

$$GG_{\sigma_{I_w},\gamma}(I_w) = P(\sigma_{I_w},\gamma)exp\{-[\delta(\sigma_{I_w},\gamma)\ |I_w|\ ]^{\gamma}\} \tag{3.6}$$

where, $-\infty < i_w \in I_w < +\infty, \gamma > 0$,

$$\delta(\sigma_{I_w},\gamma) = \sigma_{I_w}^{-1}\left[\frac{\Gamma(3/\gamma)}{\Gamma(1/\gamma)}\right]^{\frac{1}{2}} \tag{3.7}$$

and,

$$P(\sigma_{I_w},\gamma) = \frac{\gamma\cdot\delta(\sigma_{I_w},\gamma)}{2\Gamma(1/\gamma)} \tag{3.8}$$

σ_{I_w} is the standard deviation of the subband wavelet coefficients, γ is the shape parameter and, Gamma function Γ is defined as

$$\Gamma(x) = \int_0^{\infty} exp\{-y\}y^{x-1}dy \tag{3.9}$$

Suggested by [22], for most natural images the distribution of the wavelet coefficients in a subband can be described by a shape parameter γ in the range of [0.5,1]. Considering such a distribution for the Wavelet coefficients and estimating γ and σ_{I_w} for each subband, the soft threshold τ_s which minimizes the Bayesian Risk [22, 23] can be obtained by:

$$\Re(\tau_s) = E(\hat{I}_w - I_w)^2 = E_{I_w}E_{J|I_w}(\hat{I}_w - I_w)^2 \tag{3.10}$$

where $\hat{I}_w$ is $\tau_s(J)$, $J|I_w$ is $N(I_w, \sigma)$ and I_w is $GG_{\sigma_{I_w},\gamma}$. Then the optimal threshold τ_s^* is given by:

$$\tau_s^*(\sigma_{I_w}, \gamma) = arg \ \min_{\tau_s} \ \Re(\tau_s) \tag{3.11}$$

There is not a closed form solution for (5.10) and numerical calculation is used to find τ_s^*. An estimation of the value τ_s^* is concluded by setting the threshold as [22, 23]:

$$\hat{\tau}(\hat{\sigma}_{I_w}) = \frac{\hat{\sigma}_n}{\hat{\sigma}_{S_w}} \tag{3.12}$$

where $\sigma_{S_w}^2$ and σ_n^2 are variances of pure signal (S) subband wavelet coefficients and noise respectively.

3.2 Segmentation and Localization

Image segmentation is an important operation to separate objects of interest from the background in applications of automated image analysis and robotics [60, 61]. The aim of image segmentation is partitioning of the image into distinct and uniform regions regarding some property such as grey level intensity, colour, or texture. Although in the past decades image segmentation has attracted a lot of researchers in computer vision and many segmentation algorithms have been proposed [49, 59, 98], it is still attractive due to many new applications which have been emerged by recent advancements in digital imaging. Most of the segmentation methods have been developed based on maintaining some pixel properties in a neighbourhood. The two main properties of interest are *similarity* and *discontinuity* that lead to region-based [49, 57, 59, 60, 98] and boundary-based [49, 57, 59, 60, 98] segmentation methods. Basic segmentation algorithms, including boundary-based and region-based techniques, fail to produce accurate segmentation results for several applications [101]. In recent years, therefore, there has been more interest towards model-based algorithms by taking advantage of some prior information regarding the objects of interest.

Medical image segmentation and localization plays an important role in automating the process and analysis of medical images by the delineation of anatomical structures and biological structures in macro and micro levels respectively. Segmentation and localization of anatomical and biological structures are very challenging due to complex and deformable geometries and priors. Moreover, the inherently high noise and low spatial resolution of medical images make the segmentation task more challenging. The performance of image understanding, analysis, and interpretation are closely related to the segmentation performance, thus an effective segmentation is crucial. However there is always a tradeoff between the speed of the algorithm and the segmentation accuracy.

3.2.1 Region-Based Segmentation

In region-based segmentation, pixels that are segmented in a region must

- be connected based on some predefined notion of connectivity,
- satisfy a predefined notion of similarity such as having the same gray level intensity or belonging to specific range of grey level or colour intensities.

Therefore, pixels segmented in different regions are disjoint and do not share the same range of values for the similarity property.

Split & Merge and Region Growing [57, 60, 61] have been widely used for image segmentation and belong to the region-based segmentation algorithms. They both essentially maintain the similarity property, however the segmentation process is different.

Split and Merge

As its name implies, split and merge algorithm consists of two main operation: split and merge [15, 25, 57]. The basic idea is to split the image into disjoint regions with intra-

region coherency. The split & merge algorithm starts by considering the entire image as a single area of interest and can be summarized as in Algorithm 1.

Algorithm 1 Split and Merge

1: Take the entire image as a single region, so we begin with this region as area of interest.
2: **repeat**
3: **if** The area of interest satisfies the similarity constraints **then**
4: The area of interest belongs to a region in the image.
5: **else**
6: Split the area into four equal quarters. Each quarter will be considered as an area of interest.
7: **end if**
8: **until** Each area of interest satisfies the similarity constraints.
9: % The segmentation result after splitting process often contains neighbouring regions with similar properties.
10: **repeat**
11: Merge adjacent regions which satisfy similarity properties.
12: **until** No adjacent region satisfies the similarity constraints.

Region Growing

Region Growing [34, 49, 57, 59, 66, 70, 109, 122] is a basic, well-known region-based segmentation algorithm. Some starting points, seed pixels, are selected first. The seed points are usually selected manually by the user. Each region will grow around a seed point by including the neighbouring pixels that satisfy the similarity constraint to produce homogeneous regions step by step. The merging process stops when no more pixels can be merged to the region. Each pixel is merged to the region-based on the similarity of the evaluated

Algorithm 2 Region Growing

1: Start with pixel p, a selected seed point, so the area of interest P contains just pixel p.
2: Define a similarity measure $C(i, j)$ that has a high value whether pixels i and j are similar and a low value otherwise.
3: Generate a set of pixels S containing all neighbouring pixels of the area of interest.
4: **for** each pixel q of list S **do**
5: **if** if $C(p, q) > T$ **then**
6: Include pixel q to area of interest P
7: **end if**
8: Add neighbouring pixels of q to the list S which are neither a member of S nor a point in P.
9: **end for**

pixel and the region considering a threshold. The threshold determines the degree of similarity of pixels in a region, so choosing a right value for the threshold is crucial to avoid under or over segmentation. The similarity of the pixels can be evaluated based on intensity, color, texture, or any other feature. The segmentation result may contain some small regions or similar neighbouring regions. Such regions can be eliminated by merging in the postprocessing phase to generate broader regions and to reduce the number of segmented regions. Region growing algorithm can be summarized as in Algorithm 2.

The main concerns that must be considered using a region growing algorithm are to define an appropriate similarity measure $C(i, j)$ and choosing the right threshold T. The tradeoff here is selecting a very high threshold produces a region with great coherency in which pixels are very similar, however it results in an over-segmented image in which a homogeneous region will be segmented into several smaller subregions. On the other hand a very small threshold makes the algorithm more flexible which produces large regions,

however it results in an under-segmented image in which inhomogeneous regions will be segmented as a single region.

3.2.2 Boundary-Based Segmentation

In contrast with the region-based segmentation which is segmentation based on the similarity of pixel intensities, boundary-based segmentation is segmentation based on abrupt changes in pixel intensities in a neighbourhood. In this method, borders of different objects are discovered based on the discontinuity of pixel intensity values [49, 57, 59, 60, 61, 98].

Edge detection methods based on computing gradients or Laplacian of images are usually used to discriminate region boundaries [20, 21, 57, 59, 61, 86, 120]. An edge can be defined as a transition from one region to another which causes a discontinuity along the transition boundary. To detect edges in an image, most edge detectors measure the gradient intensity at each point in the image by computing the difference between pixels in the neighborhood. The higher the difference, the stronger the edge. Due to the lack of contrast and the presence of noise, detected edges as probable transitions between regions might represent incomplete borders.

3.2.3 Region-Based vs. Boundary-Based

Two basic segmentation techniques including region-based approach that evaluates the similarity property, and boundary-based approach that discovers discontinuities in pixel intensity values have been discussed. Each technique has some advantages and some shortcomings that will be briefly discussed here.

Due to intensity variations and noise in an image, segmentation algorithms developed using the boundary-based technique do not necessarily produce closed region boundaries.

Boundary-based methods might produce missing, discontinuous, and spurious edges. However isolated edge fragments can be merged and connected to form a closed contour as region boundary. Therefore boundary-based techniques can be designed to handle gaps that are caused by occlusions, to produce a closed path whereas region-based techniques usually fail to find objects that span multiple disconnected regions.

On the other hand most region-based methods are easy to implement, simple, and fairly quick. Region-based methods produce coherent regions and a closed path can be produced as the region boundary while boundary-based methods usually produce isolated edge fragments and to eliminate the gaps, postprocessing edge linking methods must be applied. Region-based techniques perform well for the segmentation of easily distinguishable objects and are fairly quick in calculations of area and volume of segmented regions. However they perform poorly to segment occlusions and similar close separate regions. Applying edge detectors is often easier than making decisions about region membership and similarity measure.

3.2.4 Line Detection

As we discussed earlier in Sec. 3.2.2, Boundary-based segmentation methods are developed based on the discontinuity property. Line detection methods that are often used to discriminate object boundaries can be grouped as boundary-based algorithms. In many applications, we do not have prior information about the general shapes of regions and objects in advance, thus we use edge detection methods to explicitly find connected sets of edge pixels. However in some segmentation problems we might have some prior information about the the general shape of the object or region. Line fitting (or more general shape fitting) approaches such as the Hough transform [64] look for the best fit of a known line to the edge data. These approaches have been widely used [5, 51, 79, 130] and are

useful for many segmentation applications. In this section, some line detection methods are presented which have been used for road detection in satellite images [53], finding cracks in concrete [117], and arteries in retinal images [81].

In general, line detection methods are grouped into two broad categories: general purpose and specific line detection approaches. However these methods can be grouped into several subsets based on their specifications

Based on how the method is applied:

- Sequential (search) methods [5].
- Parallel (filter) methods [77].

Based on what kind of image the method can be applied to:

- Binary line detection methods [44].
- Gray level line detection methods [53].

Much research has been done and several different methods have been introduced either to solve the line detection problem in general or to come with a solution for a specific application. Among line detection methods there are several extensions to the basic Hough transform approach [5, 51, 130].

Hough Transform

The most well-known line detection method which has been introduced as a general solution for line detection is the Hough Transform. This method was introduced first by Paul Hough in 1962 [64] to locate tracks in bubble-chambers. It was brought to the attention of the computer vision community by Rosenfeld [108] and was later refined by Duda and Hart

[44]. The Hough transform uses an analytical representation of a line in two-dimensional images to find straight lines by employing an accumulator as a digitized line detector. Line fitting approaches such as the Hough transform try to find the best fit of a known line to the edge data, so they do not need to connect the edge pixels.

Application Based Line Detection Methods

There are several line detection methods which have been developed to solve a specific line detection problem. Some of them are extensions of Hough transform which have been introduced to improve the Hough transform and cast it for a specific task. Among them the most important ones are:

- Generalized Hough transform [5]
- Randomized Hough transform [130]
- Randomized Generalized Hough transform [51]
- Dynamic Generalized Hough transform [79]

The others mostly are line detection filters that identify lines by rotating and convolving a filter with the original image such as:

- An active testing model for tracking roads in satellite images [53]
- A Bayesian geometric model for line network extraction [77]
- Automated Segmentation of Underground Pipe Scanned Images [117]
- A local approach for fast line detection [81]

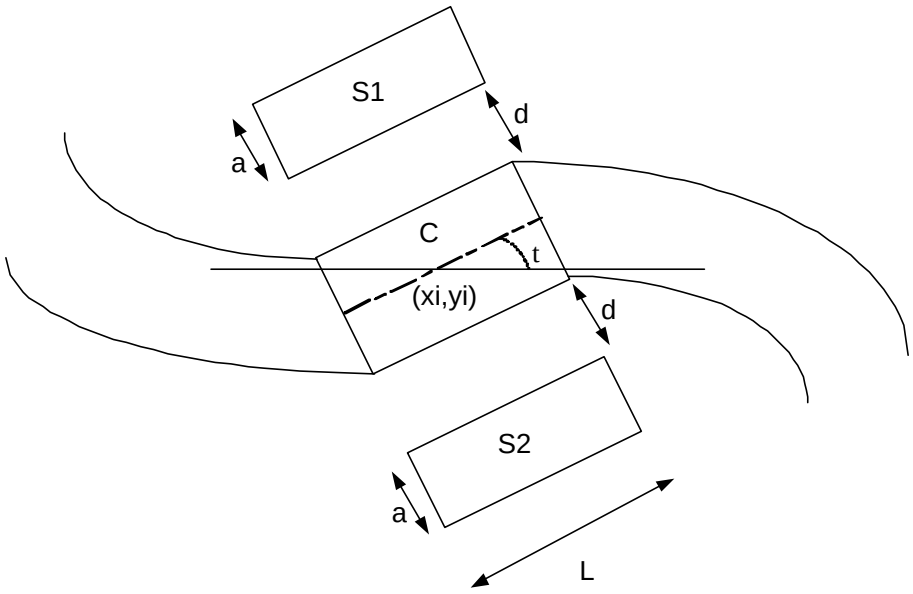


Figure 3.1: A mask for satellite network line detection consisting of a central region and two non overlapping neighbourhood regions on the opposite sides.

In these methods a straight line is represented by a central region and two non overlapping neighbourhood regions on the opposite sides. Then the response of this line detector is computed based on the homogeneity of intensities inside the center region and inhomogeneity between the center and side regions. A pixel located in the central region is more likely to belong to a line if it has

- A large gray level variations between the center and side regions.
- A homogenous background inside the center region.

As depicted in Fig. 3.1, to represent a road and background a mask composed of three regions including the central region (C) and side regions ($S1$) and ($S2$) is defined. To have a more general representation, the side regions are allowed to be located with distance d from the central one.

3.2.5 Region/Boundary Based Segmentation - Hybrid Method

The edge information can be employed as an additional criterion in conjunction with the application of the region-based methods [15, 55, 133]. Bonnin *et. al.* proposed a controlled split and merge algorithm for segmentation using detected edges [15].

In this algorithm they have used both similarity and discontinuity properties for segmentation. A region is considered homogenous if it has a small standard deviation regarding a similarity property. At the same time, pixels on an edge satisfy the discontinuity property leading to boundary-based segmentation. Therefore to decide whether a region must be split, the edge information are integrated with the similarity property. In this way, a homogenous region is explained as a region that has no edges. In the merging phase, adjacent regions will be merged if there are no edges on the common boundary.

Edge information can be used in the region growing method as well to decide whether a pixel must be aggregated to a region or not. Running into an edge indicates that the growing process must be stopped, the pixel must not be combined with the region, and we have reached the region boundary. *Kara et. al.* [1] proposed an algorithm in which pixels having a low gradient value will be aggregated to the growing region in each iteration. In the method proposed by *Gambotto* [52], the growing process will be stopped using edge information assuming that the gradient has high values over region boundaries. *Xiaohan et. al.* [129] proposed a method based on similarity property considering the weighted sum of the pixel gradient, and the pixel-region contrast. The pixel will be merged to the region for small values of the function while the growing will be stopped for large values.

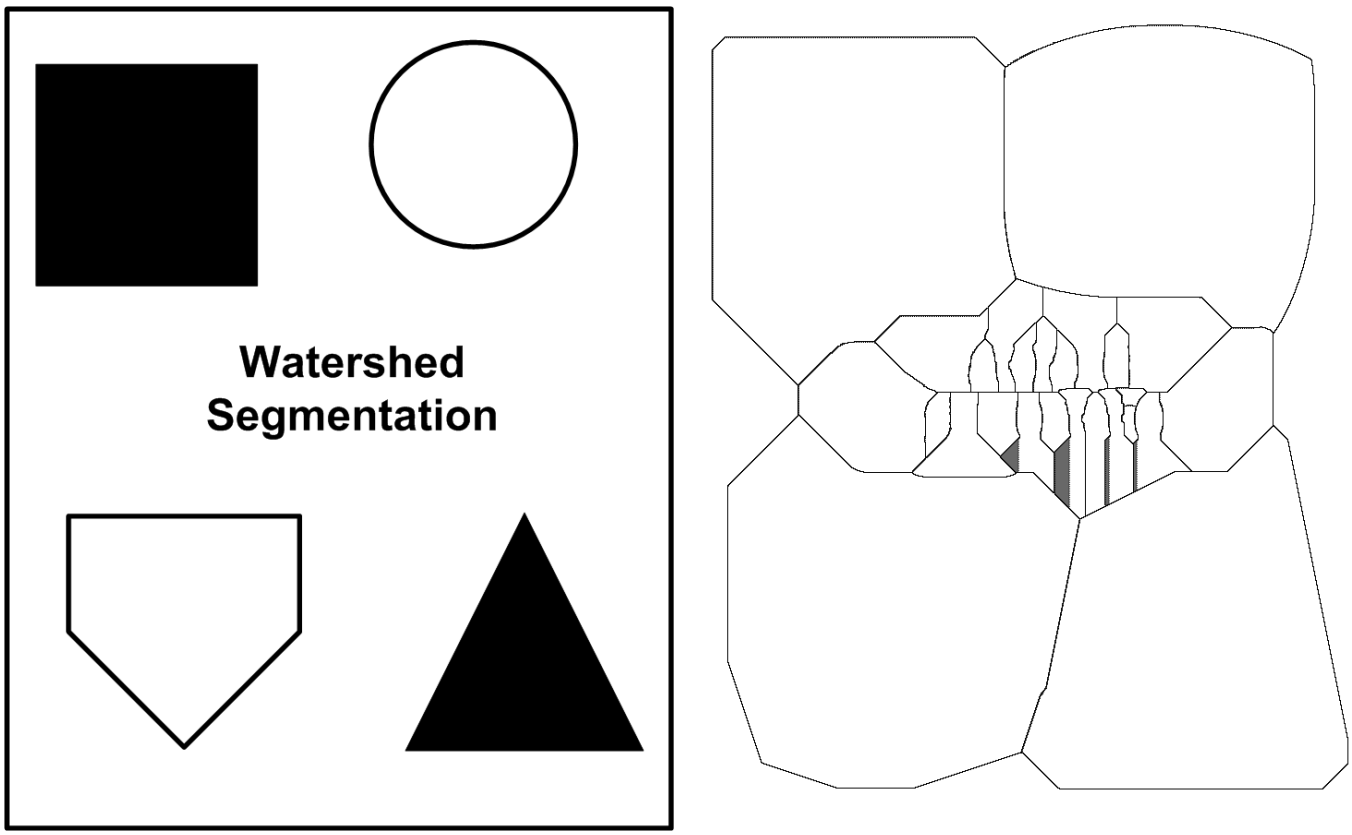


Figure 3.2: (Left) Original image. (Right) Watershed segmentation of the left panel.

3.2.6 Mathematical Morphology - Watershed

I use watershed segmentation in one of the proposed cell segmentation methods in chapter 6, so the watershed segmentation is briefly explained here. The watershed transform has been widely used for image segmentation. It was originally proposed by Lantuejoul and Digabel [38, 78] and was improved later by Beucher and Lantuejoul [10]. The watershed method partitions the image into disjoint homogeneous regions regarding some similarity property while it also provides regions boundaries, thus it can be grouped as a hybrid region/boundary based segmentation approach [11, 59, 112, 124].

The idea underlying the watershed method is taken from geography. Considering a landscape area which is flooded by water, watershed lines divide the regions of rain fall attractions [57, 59]. Alternatively the landscape can be considered being plunged in a lake where some holes represents the local minima. Starting at local minima, catchment basins will fill up with water. Dams are constructed where water coming from different catchment basins comes across. This process will be stopped as soon as the water reaches the highest level in the landscape. Thus the landscape is partitioned by constructed dams know as watershed lines into catchment basins or regions.

This process has been used to simulate image segmentation in the field of image processing. Watershed segmentation is a computational algorithm which segments the region boundaries by filling up the local minima of a grey level image recursively. In the same way as topographical watershed lines, regions are filled from the bottom. Dams are constructed to separate regions where regions reach a level that they would merge. Therefore watershed lines or dams become the region boundaries where catchment basins represent the regions [57, 59]. There are two possible approaches to implement the watershed:

- Locating catchment basins first; As a result, watershed lines are the complement set of catchment basins.
- Partitioning of the image into catchment basins; Locating the watershed lines by boundary detection.

Applying the watershed segmentation, the image will be labelled as follows:

- Pixels in a given catchment basin have the same label.
- Pixels of all watershed lines have the same unique label which is different from the labels of the catchment basins.

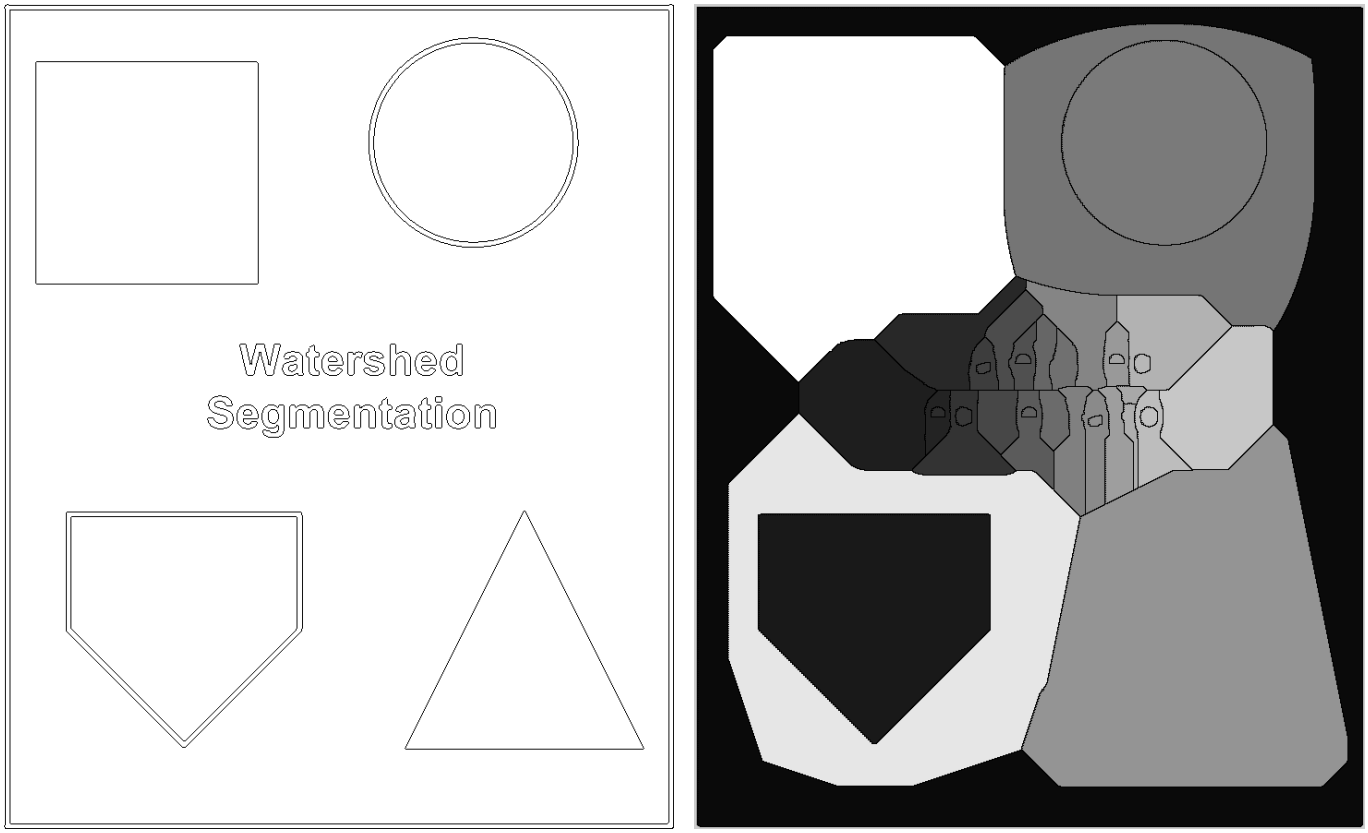


Figure 3.3: (Left) Detected edges applying Sobel kernel to the image depicted in (Fig. 3.2(Left)) (Right) Watershed segmentation of the left panel.

Watershed lines indicates region boundaries, thus in image segmentation task it is desired to represent watershed lines as sets of pixels with grey level discontinuity, thus the watershed algorithm can be applied to the gradient image [57, 93] in place of the original image as depicted in Fig. 3.3.

3.2.7 Thresholding Method

I use different thresholding methods including Otsu clustering method [97] in the proposed cell segmentation/localization approaches in chapter 6. To familiarize readers who are unfamiliar with thresholding, this method will be briefly explained here.

The goal of image segmentation is dividing an image into separate regions or boundaries corresponding to different objects. In region-based segmentation, regions are segmented by identifying pixels with similar properties while in boundary-based segmentation, contours are segmented by identifying pixels with different properties. Gray level intensity is the simplest property to test the similarity of pixels in a region. Therefore to segment an image to dark and bright regions, pixel intensities can be thresholded.

One of the most popular techniques of gray level image segmentation is the histogram thresholding method [21, 57, 60, 102, 113]. Image thresholding converts a grey-level image to a binary image by setting to zero all pixels with intensities below some threshold and turning to one all pixels with intensities above that threshold. Objects and background of a gray level intensity image that are respectively associated with peaks and valleys of the intensity histogram can be separated using a threshold τ.

The thresholding method is a fast and simple point processing method, however it only performs well where there is sufficient contrast between objects and background. Two extensions of thresholding method known as global and local thresholding [21, 57, 60, 102, 113] are discussed here.

Global Thresholding

Assume $I(x, y)$ is a gray level image and $b(x, y)$ is its thresholded image considering a global threshold τ. We can write

$$b(x,y) = \begin{cases} 1 & I(x,y) \geq \tau(I) \\ 0 & I(x,y) < \tau(I) \end{cases} \tag{3.13}$$

where the threshold value $\tau(I)$ is only a function of the image gray level intensities. The major shortcoming of global thresholding is that there is no guarantee that the segmented pixels as foreground objects by the thresholding process be contiguous. The reason is that only the intensity is considered in the thresholding method and the relationships between the pixels will not be taken into account. As a result, some pixels that are not part of a region might be considered to be in the region and also some pixels that belong to the region, for example pixels close to the boundaries of an object, might be segmented as background.

Local Thresholding

The threshold value τ can vary and be set across the image locally in place of setting a global threshold. The advantage of the local thresholding is handling the illumination variations across the image. Illumination variations might cause some parts of a single object look brighter and some parts darker, or having a brighter background in one part and darker on the other part of image. The image b is obtained by local thresholding of the image I as

$$b(x,y) = \begin{cases} 1 & I(x,y) \geq \tau(I, f(x,y)) \\ 0 & I(x,y) < \tau(I, f(x,y)) \end{cases} \tag{3.14}$$

where threshold $\tau(I, f(x, y))$ is a function of image gray level intensities I and a function f such as sample mean computed on a local spatial neighbourhood centered at (x, y).

Selecting a Global Threshold

The following threshold selection methods are described to set a global threshold, however these methods can be applied to set local thresholds computing a function such as sample mean locally and considering local histograms.
There are two general approaches to select a threshold as follows.

1. Manual or heuristic approach for selection of the global threshold τ by visual inspection of image histogram,

2. Automatic approach for selection of the global threshold.

Manual Selection of a Threshold

Objects and background of a gray level intensity image are associated with distinct peaks of the intensity histogram. To set a threshold τ in a manual approach for separating objects and background, the image histogram must be visually inspected to find two distinct modes and then find the valley between them. This method performs well for smooth histograms with two significant modes, however it fails for noisy histograms with multiple peaks. In such cases we can filter the image to obtain a smooth histogram, find its modes and set a threshold between the modes.

Automatic Selection of a Threshold

Another approach is to set a threshold automatically. Assume R_B is the range of background intensity values and R_F is that of foreground. The ideal threshold τ separates these intensity ranges completely. If R_B and R_F are distinct, the threshold would be set in the valley separating two ranges. However, background and foreground intensity ranges are not usually distinct and share some overlapping areas. So wherever we set the threshold τ,

some background pixels will be classified as foreground and some foreground pixels will be classified as background. Therefore an appropriate threshold can be selected by minimizing the classification error where a pixel is classified in a wrong class.

The threshold can also be set to minimize the overlapping area for each region, i.e., by minimizing the tail of each region's pdf that lies in the other region's side. To do this let the intensity levels in the background and foreground regions be considered as two clusters. A threshold can be set by

$$\begin{aligned} &if\ I(x,y) \geq \tau \Rightarrow |I(x,y) - \mu_F| < |I(x,y) - \mu_B| \\ &if\ I(x,y) < \tau \Rightarrow |I(x,y) - \mu_F| > |I(x,y) - \mu_B| \end{aligned} \tag{3.15}$$

where μ_B and μ_F are background mean and foreground mean respectively. Obviously τ is

$$\tau = \frac{\mu_B + \mu_F}{2} \tag{3.16}$$

The question here, that will be answered in the next section, is how to calculate μ_B and μ_F where we do not know the background and foreground regions. If we knew them, thresholding would not be needed to segment the image in the first place.

K-means Clustering to Select a Threshold

K-means is a well-known method and has been widely used for clustering [71, 85, 87]. To select the threshold τ to satisfy (3.15), K-means can be started with a randomly selected initial value for threshold and cluster the pixels to background and foreground. K-means is summarized in Algorithm 3.

Otsu Clustering to Select a Threshold

Another method to minimize the overlapping area of the background and foreground clusters is called Otsu's method [97]. Otsu set the threshold to generate condensed clusters.

Algorithm 3 K-means clustering

1: Select an initial threshold τ_0.
2: $\tau = \tau_0$
3: **repeat**
4: Segment image to two sets of pixels including foreground F and background B using threshold τ.
5: % Compute sample means for each set.
6: $\mu_F = mean(F)$
7: $\mu_B = mean(B)$
8: $\tau_{old} = \tau$
9: $\tau = \frac{1}{2}(\mu_F + \mu_B)$
10: **until** $\Delta\tau = \tau - \tau_{old} < \epsilon$.

Using this method a gray level image can be clustered to foreground and background clusters by minimizing the intra-class variance that is the weighted sum of the variance of each cluster and is defined as

$$\sigma^2_{intra}(\tau) = n_F(\tau) \cdot \sigma^2_F(\tau) + n_B(\tau) \cdot \sigma^2_B(\tau) \tag{3.17}$$

where $n_B(\tau)$ and $n_F(\tau)$ are the number of pixels in the background and the foreground clusters, $\sigma^2_B(\tau), \sigma^2_F(\tau)$ and $\sigma^2(\tau)$ are background, foreground, and intra-class variance respectively, as a function of the threshold τ. In Otsu's method the intra-class variance is computed for all intensity levels in histogram and the threshold with the minimum intra-class variance is selected as the desired threshold. Otsu's method is summarized in Algorithm 4. The computation of $\sigma^2_B(\tau)$ and $\sigma^2_F(\tau)$ are very expensive, since τ can accept all possible intensity levels in the histogram. To simplify the computational cost assume

Algorithm 4 Otsu clustering

1: **for** i=1:H **do**
2: $\tau = h_i$ (h_i is a typical intensity level in histogram)
3: Cluster the histogram to two sets of pixels including foreground F and background B using threshold τ.
4: Count the number of pixels in each cluster, n_F and n_B.
5: % Compute sample variances for each cluster.
6: Compute $\sigma_F^2(\tau)$
7: Compute $\sigma_B^2(\tau)$
8: $\sigma_{intra}^2(\tau) = n_F(\tau) \cdot \sigma_F^2(\tau) + n_B(\tau) \cdot \sigma_B^2(\tau)$
9: $V_i = \sigma_F^2(\tau)$
10: **end for**
11: Select the τ that produces minimum $\sigma_{intra}^2(\tau)$.

that the combined distribution's variance is σ_{Tot}^2 and we have

$$\begin{aligned} \sigma_{inter}^2 &= \sigma_{Tot}^2 - \sigma_{intra}^2 \\ &= n_B(\tau) \cdot (\mu_{Tot} - \mu_B(\tau))^2 + n_F(\tau) \cdot (\mu_{Tot} - \mu_F(\tau))^2 \end{aligned} \tag{3.18}$$

where σ_{inter}^2 is the inter-class variance and μ_{Tot} is the histogram mean. Considering that

$$\mu_{Tot} = n_B(\tau) \cdot \mu_B(\tau) + n_F(\tau) \cdot \mu_F(\tau) \tag{3.19}$$

and replacing (3.19) in (3.18) we have

$$\sigma_{inter}^2(\tau) = n_B(\tau) \cdot n_F(\tau) \cdot (\mu_F(\tau) - \mu_B(\tau))^2 \tag{3.20}$$

In this way in place of minimizing the intra-class variance to find the optimal threshold we maximize the inter-class variance that depends only on the difference of cluster means which

is much easier to compute than the sum of weighted variances. Moreover the computations of $n_B(\tau)$, $n_F(\tau)$, $\mu_B(\tau)$, and $\mu_F(\tau)$ are related when changing the threshold τ from one value to another and can be updated as some pixels move from one cluster to another. Assume n_{New} is the overall number of pixels that switch from one class to another when the threshold is increased, so we can write

$$\begin{aligned} n_B(\tau+1) &= n_B(\tau) + n_{New} \\ n_F(\tau+1) &= n_F(\tau) - n_{New} \end{aligned} \tag{3.21}$$

As a result the inter-class variance can be updated each time the threshold is changed using recursive relations as

$$\begin{aligned} \mu_B(\tau+1) &= \frac{\mu_B(\tau)\cdot n_B(\tau) + n_{New}\cdot\tau}{n_B(\tau+1)} \\ \mu_F(\tau+1) &= \frac{\mu_F(\tau)\cdot n_F(\tau) - n_{New}\cdot\tau}{n_F(\tau+1)} \end{aligned} \tag{3.22}$$

The fast implementation of Otsu clustering is summarized in Algorithm 5.

Algorithm 5 Fast Otsu

1: **for** i=1:H **do**
2: $\tau = h_i$ (h_i is a typical intensity level in histogram)
3: Update the number of pixels in each cluster, $n_F(\tau)$ and $n_B(\tau)$.
4: Update sample means for each cluster, $\mu_F(\tau)$ and $\mu_B(\tau)$.
5: $\sigma^2_{inter}(\tau) = n_B(\tau)\cdot(\mu_{Tot} - \mu_B(\tau))^2 + n_F(\tau)\cdot(\mu_{Tot} - \mu_F(\tau))^2$
6: **end for**
7: Select the τ that produces maximum $\sigma^2_{inter}(\tau)$.

3.2.8 Mean Shift Procedure

Mean shift is a non-parametric estimator of gradient of a density function and has been used to find clusters with arbitrary shapes [27, 32, 50]. Fukunaga and Hostetler [50] introduced the mean shift estimate of density gradient and developed an iterative procedure to find the peaks of density gradient. In the mean shift algorithm a nonparametric kernel density estimation method is used to find local modes of the density. The idea is to move in the direction of the gradient of the density of the data until converging to density modes.

Assume we have a set of N points in k-Dimensional Euclidean space R^k as $\{x_i,\ i \in [1:N]\}$ and $W(x)$ is a kernel with window radius h. In an arbitrary point x, the multivariate kernel density estimate is [116]

$$\hat{P}(x) = \frac{1}{Nh^k} \sum_{i \in [1:N]} W\left(\frac{x - x_i}{h}\right) \tag{3.23}$$

The estimate of the density gradient can be replaced by the gradient of the kernel density estimate where using a differentiable kernel

$$\hat{\nabla} P(x) \equiv \nabla \hat{P}(x) = \frac{1}{Nh^k} \sum_{i \in [1:N]} \nabla W\left(\frac{x - x_i}{h}\right) \tag{3.24}$$

Let $W(x)$ be a Epanechnikov kernel [44] which is the optimum kernel based on minimum mean integrated square error (MMISE)

$$W(x) = \frac{3}{4}(1 - x^2),\ \text{for } |x| \leq 1 \tag{3.25}$$

$W(x)$ is equal to zero for $|x| > 1$. By replacing $\hat{P}(x)$ in (3.24) we have

$$\hat{\nabla} P(x) = \hat{P}(x) \frac{k+2}{h^2} \left(\frac{1}{N_x} \sum_{x_i \in H_h(x)} (x_i - x)\right) \tag{3.26}$$

The second term of (3.26) is known as mean shift

$$S_h(x) \equiv \frac{1}{N_x} \sum_{x_i \in H_h(x)} (x_i - x) = \frac{1}{N_x} \sum_{x_i \in H_h(x)} x_i - x \tag{3.27}$$

The mean shift vector points towards the direction of the gradient of the density estimate at x, thus it has the direction of the maximum increase in the density and leads to a density mode (local maximum).

The mean shift algorithm is a natural extension of the discontinuity preserving smoothing algorithm and has been used for image segmentation [32]. Applying the mean shift algorithm, each pixel will converge to a local mode (of the density function) in its neighbourhood. The segmentation method using mean shift algorithm is summarized in Algorithm 6 assuming $\{x_i,\ i \in [1:N]\}$ is a set of the original k-Dimensional input points, $\{s_i,\ i \in [1:N]\}$ is a set of converged points, and $\{l_i,\ l_i \in [1:L]\ \&\ i \in [1:N]\}$ is a set of labels.

Algorithm 6 Mean Shift Algorithm

1: **for** i=1:N **do**
2: %Application of the mean shift algorithm to each x_i converges to s_i.
3: $s_i = S_h(x_i)$
4: **end for**
5: Cluster converged points s_i in density domain to obtain L clusters $\{R_c|\ c \in [1:L]\}$.
6: $l_i = \{c|\ s_i \in R_c,\ c \in [1:L]\ \&\ i \in [1:N]\}$
7: Remove spatial regions smaller than T pixels.

3.3 Background Modelling and Estimation

There is an increasing number of image analysis problems in which modelling or estimation of a background image is crucial. Object segmentation, recognition, and tracking are important problems that often demand for background modelling. In all of these problems, the interest is separating or decoupling foreground and background, where the definition of

the foreground objects might vary from one application to another. Therefore to solve these problems with respect to their specific applications, modelling or estimation of background might be mandatory. We might either need to model the background and classify image pixels to foreground and background, or estimate the background image and subtract it from the original one to obtain the foreground. In this section background modelling methods are briefly reviewed.

Although most televised videos involve frequent scene cuts and camera motion, a great deal of imaging such as medical and biological imaging are based on a fixed camera which yields a static background and a dynamic foreground. Moreover, in most tracking problems the dynamic foreground is of interest, hence an accurate estimation of background is desired. Removing the estimated background leaves us with foreground on a plain background. The estimated background might be composed of random temporal noise, temporal illumination variations, spatial distortions caused by CCD camera pixel non-uniformities, and stationary or quasi-stationary clutter and background structures.

In the ideal case, the background image contains only stationary objects. These objects have no motion and are stationary such as walls, ground, and tables in a room. However a background scene might not be stationary. There are more challenging cases in which the background image can be composed of stationary, quasi-stationary and non-stationary objects. The stationary objects contain static features while dynamic features belong to the non-stationary objects.

Moreover quasi-stationary objects usually do not leave the scene and might have a periodic, repetitive or random motion such as swaying trees and rippling water. The state of the objects in the background might also change from stationary to non-stationary or vise-versa over time. Such objects may have variable dynamics, enter, stay, and leave the scene. A typical scenario in monitoring and traffic control [2, 111, 115]: a car enters to the visual

field of the surveillance camera as a moving object, stops with the red light and its state changes to stationary, the light turns to the green, the car moves and leaves the scene while its state changes again from a background to a foreground object. Furthermore the appearance of static background pixels might change over time due to illumination changes, CCD camera spatial noise and random temporal noise. For all of these situations an adaptive background estimation is needed to model the dynamic changes in the background.

There are different methods for background estimation using different image features at each pixel location. In most of them spectral features of each pixel representing gray level intensity or colour information of the pixel have been used to model the background [45, 65, 72, 126]. Background estimation methods can be divided into different groups based on using gray level (or colour) intensities, spatially, temporally or sptio-temporally. Some of them have used spatial features to model the local structures of the background image [73, 83, 100]. Methods which employ spatial and spectral features have a good performance when the background image consists of stationary objects with static features but they demonstrate a poor performance when the background image consists of non-stationary objects with dynamic features.

Temporal features play a significant role in modelling non-stationary objects and frame by frame changes of the background [63, 82, 118, 126]. Among the methods which use temporal features, Gaussian mixture models have been widely used and have performed well to estimate non-stationary temporal background pixel distributions [48, 62, 118]. Different extensions of Gaussian mixture models have been introduced to improve their performance and reduce the running time [69, 80, 131, 132].

The most basic background estimation approach estimates the background by averaging the previous N frames. The averaging method performs well for stationary background and moving foreground. An adaptive version of this method which recursively computes

the background image has been used in [126]. Extensions of this method use Kalman and infinite impulse response (IIR) filters to model each pixel [72, 75, 91, 106]. In general the basic method is not effective if the background consists of non-stationary elements or foreground does not move continuously.

Statistical background methods have been used widely to model image background. A simple effective method is using a Gaussian distribution to model each pixel [14, 36, 46, 104, 126]. This method is not robust where background temporal variations are multi-modal. As a natural extension of Gaussian distribution, Gaussian mixtures have been used to accurately model multi-modal temporal variations of background caused by illumination changes and motion of non-stationary elements [62]. In this method each pixel is modelled using a Gaussian mixture. In [48] Friedman and Russell used a multi-modal Gaussian to classify each pixel into one of the predefined classes including road, vehicle and shadow. They adaptively updated the Gaussian parameters to model the background changes in their method.

Among Gaussian mixture methods to model the background, the one proposed by Stuffer and Grimson for surveillance system [118] is the most sophisticated and has been considered as the standard Gaussian mixture formulation since then. Based on the variance and persistence of each Gaussian in the mixture they classify each Gaussian and in turn its corresponding pixels to the background and foreground. Moreover instead of using an EM algorithm to solve the Gaussian mixture problem, they use an online K-means approximation to classify every new pixel to one of the K Gaussian distributions.

Different extensions to this algorithm have been introduced. KaewTraKulPong and Bowden [69] used another method to update the Gaussians and detect the shadows. Zivkovic [132] used an algorithm to update both the parameters and the number of Gaussians in the mixture. Lee [80] proposed an extention to Stuffer and Grimson to improve the

modelling convergence and accuracy. To speed up the parameter learning of the Gaussians and to maintain the stability and adaptability of the system he used an incorporation of EM and recursive filter for learning the Gaussians parameters. Heikkila and Pietikainen [63] introduced the using of the Local Binary Pattern (LBP) as discriminative texture features to estimate the background temporal distributions in the framework of Staffer and Grimson. Their method is an efficient texture based approach which can tolerate illumination changes due to the invariant attribute of LBP features.

To avoid the estimation of Gaussian parameters in the Gaussian mixture model, Elgammal *et. al* [45] proposed a nonparametric method to model the background. This is a generalized kernel-based distribution to estimate the background colour distributions while the background distributions are estimated directly from the data without preassumptions about their distribution functions.

Local texture features, local and global spatial statistics and spatial statistical models such as Markov Random Field (MRF) and Hidden Markov Model (HMM) recently have been used to improve the background estimation [73, 100]. Kato *et. al* [73] used a HMM to model the background and adapt to spatial illumination changes for traffic monitoring system. Paragios and Ramesh [100] used a combination of Gaussian and MRF to model the static background and robust foreground segmentation. Although spatial structures improve the static background estimation and shadow modelling, they demonstrate a poor performance to model the non-stationary background elements at pixel level.

3.4 Tracking

Assume $z_k = \{z_k^m |\ m \in [1, M_k]\}$ and f_k are a measurement set and system state, respectively at time k. Target tracking can be defined as the estimation of the present state

of a target f_k such as position and velocity based on the target's previous state f_{k-1} and some measurements related to present features of the target z_k as depicted in Fig. 3.4. The increasing number of analog surveillance systems and the advances in digital imaging over time are the main motivations toward systems that are capable of tracking a large number of targets. The vast amount of measured data and the presence of clutter objects are the major source of false measurements that make the tracking task very difficult. Increasing the number of targets will exponentially increase the computational complexity of the problem. Moreover when the goal is to track a number of similar targets with the same features as depicted in Fig. 3.5, the association problem becomes more complex. One of the most important and difficult problems in computer vision and image processing is tracking targets in image sequences [115]. There are numerous applications such as video compression, traffic control, medical diagnosis, living biological observations, and drug discovery [2, 9, 26, 67, 94, 103, 107, 111, 115] in which the main task is target tracking. New tracking applications have been introduced which are motivated by advances in digital imaging and image processing algorithms.

Although several approaches for tracking targets in image sequences have been introduced and developed [13, 115], the growing number of new applications with fundamental differences in their nature demand new application-based approaches. Target tracking in image sequences can be divided into two major categories, based on what is supposed to be tracked over time:

1. *Partitioning-based region tracking*: In this category, we have different regions which cover the background, touch each other and deform over time, hence the problem is partitioning the image by segmenting the boundaries of these regions and tracking the deformation of boundaries over time.

2. *Association-based object tracking*: In this category, we have well-separated objects

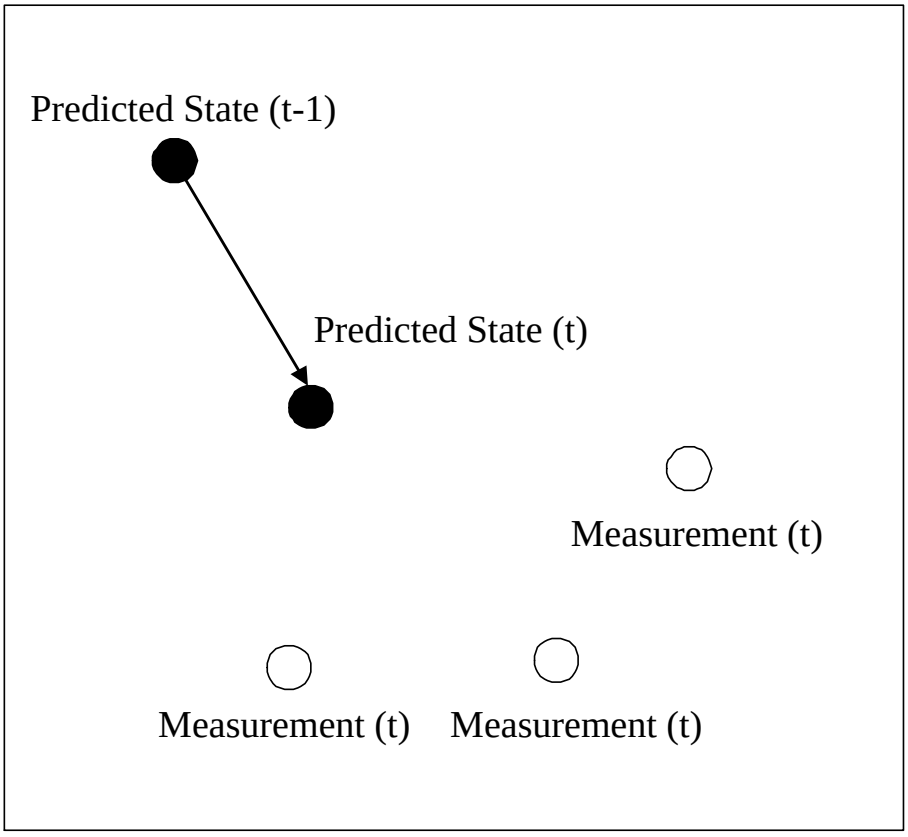


Figure 3.4: Single target tracking: the estimation of the present state of a target f_k such as position and velocity based on the target's previous state f_{k-1} and some measurements related to present features of the target z_k.

which are recognizable from the background, thus the problem is detecting these objects in the image frames and association of the detected objects over time.

The main focus of this research is on the association-based tracking where the goal is tracking a number of similar targets with the same features. The tracking problem is simplified to object detection if we track a single target in a clean background, however by increasing the number of targets the tracking problem turns to a very complex problem consisting of two stages: i) detection, and ii) association of detected objects. Usually the association problem becomes the dominant and the most challenging stage. It should be pointed out that by increasing the number of targets, the complexity of the association doesn't increase linearly but exponentially, hence the cost of tracking n targets is not equal to n times the cost of tracking one object. In the rest of this section, some of the tracking methods are explained briefly.

3.4.1 Nearest Neighbour

The nearest neighbour is a simple straight forward estimation algorithm [114, 115]. In this method the measurement that is closest to the predicted measurement is selected as the correct present state as depicted in Fig. 3.6. Nearest neighbor standard filter (NNSF) is one of the most common methods that has been widely used.

In cases where we have false measurements that are not originated from the target, the closest measurement might be a false measurement, in which case NNSF uses the false measurement (the closest measurement for this scenario) to update the target state.

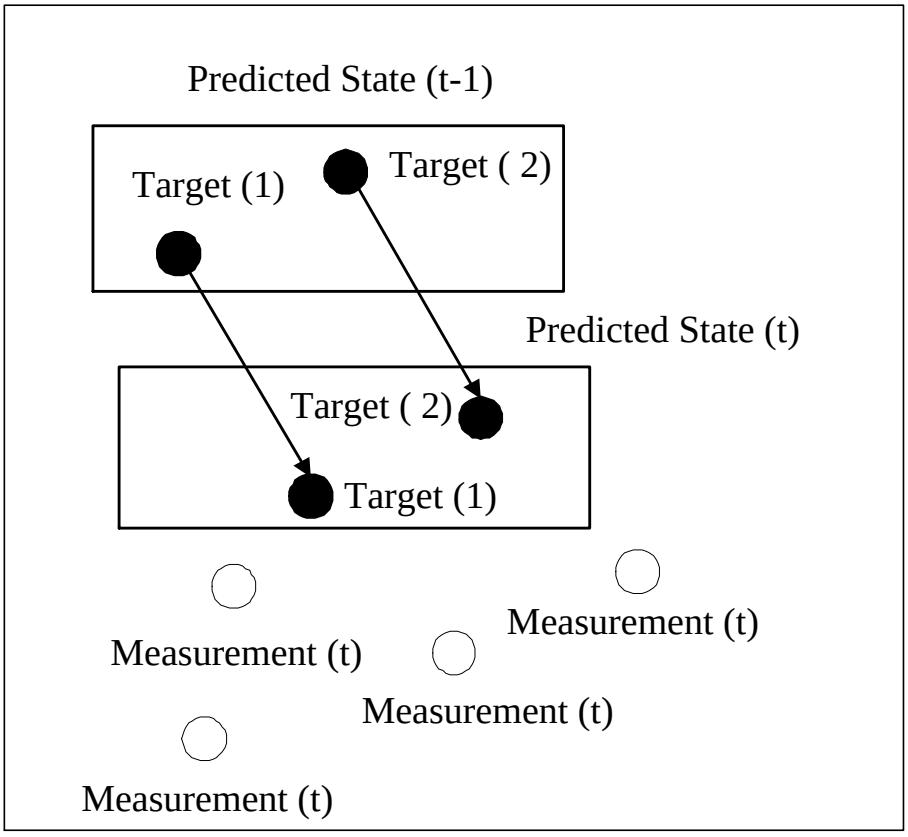


Figure 3.5: Multi-target association: tracking a number of similar targets with the same features. There is four measurements, at least two of them are false alarms, to be used to update the present state of two targets.

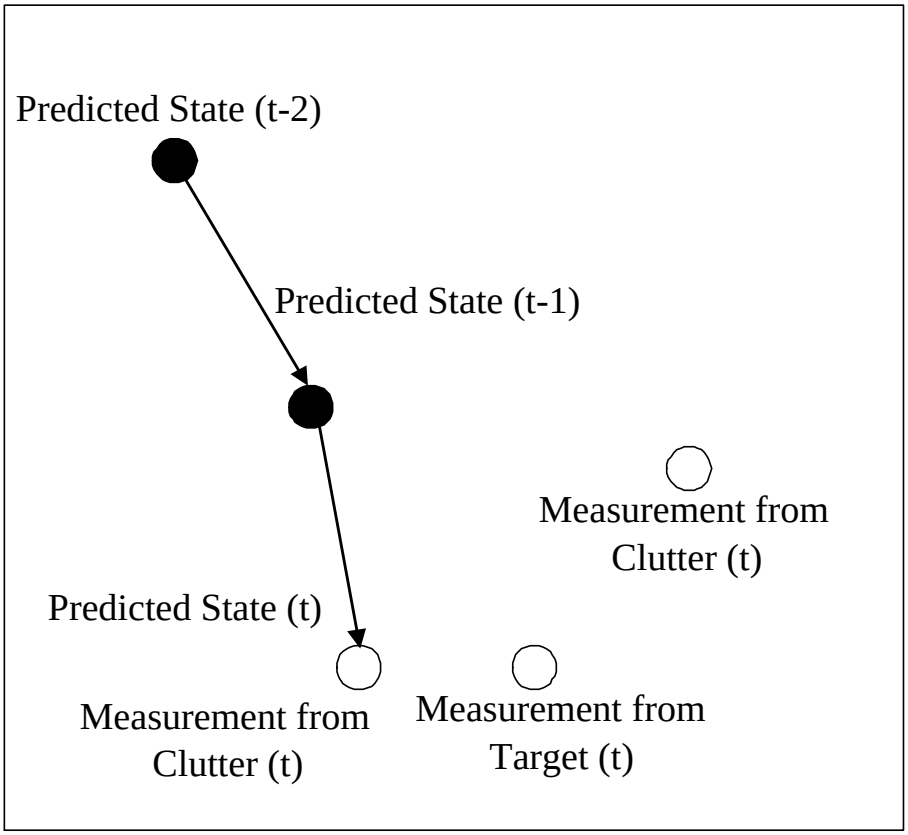


Figure 3.6: Nearest Neighbour Tracking

3.4.2 Recursive Bayesian Estimation

Let $z_k = \{z_k^m |\ m \in [1, M_k]\}$ and f_k be a measurement set and system state, respectively at time k. The optimum answer to the association problem is the maximum a posteriori estimation of $F_{1:K}$

$$\hat{F}_{1:K} = arg\ \left\{ \underset{F_{1:K}}{max}\ \ P(F_{1:K} |\ Z_{1:K}) \right\} \tag{3.28}$$

where $F_{1:K}$ is a possible hypothesis of the K-frame association (system state) and $Z_{1:K}$ is a measurement set over time $[1 : K]$. Recursive Bayesian estimation has been widely used to solve (3.28). To manage the dimensionality and so the complexity of the tracking problem, the recursive Bayesian estimation depicted in Fig. 3.7 employs a recursive strategy to estimate the present state of each target. Therefore the tracking problem (3.28) is simplified to

$$\hat{f}_k = arg\ \left\{ \underset{f_k}{max}\ \ P(f_k |\ Z_{1:k}) \right\} \tag{3.29}$$

In this way, recursive Bayesian estimation sequentially estimates the marginal distribution $P(f_k |\ Z_{1:k})$ given the measurements $Z_{1:k}$. Considering $P(f_{k-1})$ as the prior state density and $P(f_k |\ f_{k-1})$ as the system dynamics we have

$$P(f_k) = \int P(f_k |\ f_{k-1}) P(f_{k-1}) df_{k-1} \tag{3.30}$$

Assume the state density $P(f_k)$ is a *Markov Process*, so it depends only on the state density in the previous time step $P(f_{k-1})$. The state estimate is updated by incorporating the new measurement z_k at each time step k. Considering that the state *a posteriori density* is

$$P(f_k |\ Z_k) = P(f_k |\ z_k, Z_{k-1}) \tag{3.31}$$

using Bayes' rule we can write

$$P(f_k |\ Z_k) = \frac{P(z_k |\ f_k, Z_{k-1}) P(f_k |\ Z_{k-1})}{P(z_k |\ Z_{k-1})} \tag{3.32}$$

and assuming that measurements are independent

$$\begin{aligned} P(f_k|\ Z_k) &= \lambda \cdot P(z_k|\ f_k, Z_{k-1}) P(f_k|\ Z_{k-1}) \\ &= \lambda \cdot P(z_k|\ f_k) P(f_k|\ Z_{k-1}) \end{aligned} \tag{3.33}$$

where $P(f_k|\ Z_{k-1})$ is the state *a priori density*, $P(z_k|\ f_k)$ is the likelihood of measurement z_k given the state f_k, and λ is a normalizing constant. The recursive Bayesian estimation is summarized in Fig. 3.7.

3.4.3 Joint Probabilistic Data Association

Recall the recursive Bayesian estimation (3.33)

$$P(f_k|\ Z_k) = \lambda \cdot P(z_k|\ f_k) P(f_k|\ Z_{k-1})$$

In a special case where conditional probabilities are linear and Gaussian

$$\begin{aligned} P(f_k|\ f_{k-1}) &\equiv N(f_k, A_k, Q_k), \\ P(z_k|\ f_k) &\equiv N(f_k, H_k, R_k) \end{aligned} \tag{3.34}$$

the recursive Bayesian estimation can be computed in a closed form. In this case we have

$$\begin{aligned} f_k &= A_k f_{k-1} + w_k, \\ z_k &= H_k f_k + v_k \end{aligned} \tag{3.35}$$

where w_k and v_k are independent zero mean Gaussian random noise. Q_k and R_k are respectively their associated covariance matrices.

Consider a linear Gaussian model for system dynamics and measurements; Joint Probabilistic Data Association (JPDA) has been widely applied to solve (3.28) for multi-target tracking [6, 31, 114] using the closed form of recursive Bayesian estimation (3.33). JPDA assumes the number of targets T to be known with the following constraints:

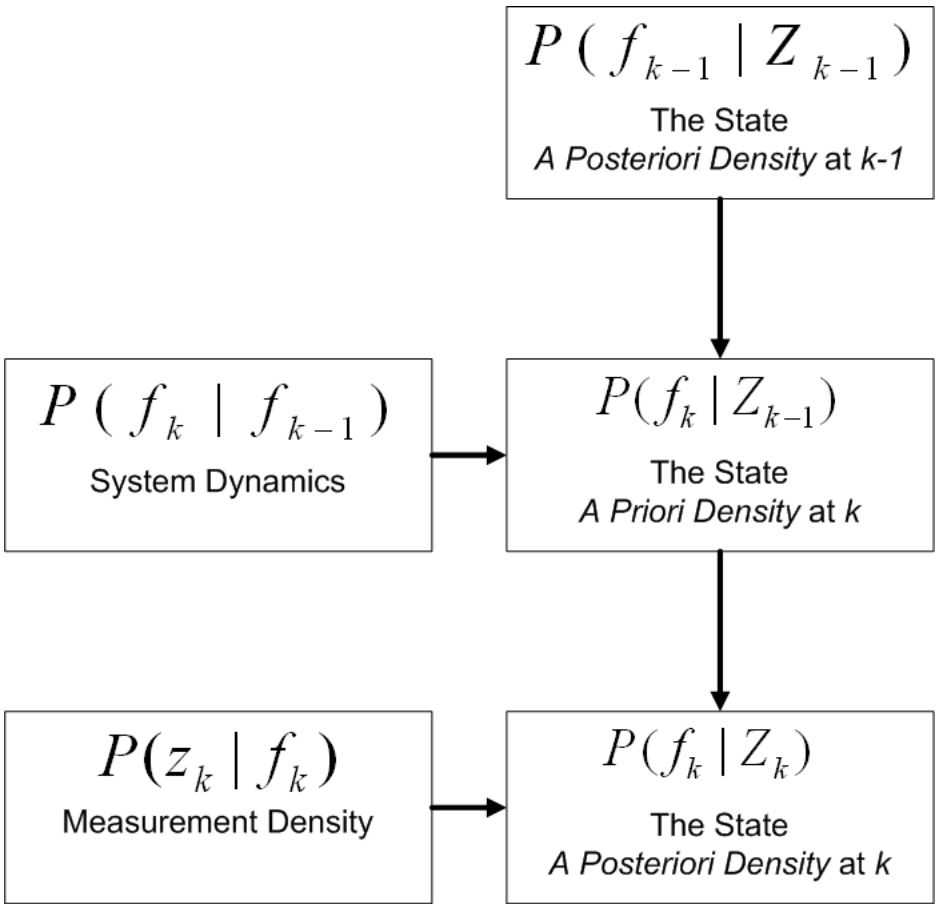


Figure 3.7: Recursive Bayesian estimation is widely used to estimate $P(f_k|\ Z_k) = \lambda \cdot P(z_k|\ f_k)P(f_k|\ Z_{k-1})$

- Each measurement originates from only target or clutter.
- A measurement can be associated to at most one target.
- At most one measurement can be associated to a target.

To make the association problem tractable, JPDA reduces the number of possible association hypotheses using a gating strategy and keeps a reasonable subset of them known as valid association hypotheses. Only the measurements that fall inside the validation gate of each target are kept as valid measurements for that target and the rest are not considered as association candidates. Therefore, JPDA associates only the validated measurements at time k to the targets by searching in a reduced hypothesis space

$$\{f_k^h |\ h = 1, 2, ...\} \tag{3.36}$$

for the best member

$$\hat{f}_k = f_k^{\hat{h}} \ \text{ where } \ \hat{h} = arg\ \left\{ \underset{h}{max}\ \ P(f_k^h | Z_{1:k}) \right\} \tag{3.37}$$

as the solution. The first term of (3.33), $P(z_k |\ f_{k,j})$, is the likelihood of measurement z_k given hypothesis $f_{k,j}$ and is computed in standard JPDA as

$$P(z_k |\ f_{k,j}) = \left[\beta_j^0 + \sum_{m \in [1, M_k]} \beta_j^m \cdot P(z_k^m |\ f_{k,j}) \right] \tag{3.38}$$

where β_j^m is the marginal posterior probability of associating measurement m to target j so that β_j^0 is the probability of no measurement to be associated to target j. The second term of (3.33) is a prediction step where Kalman filter is employed by JPDA to model the association uncertainties assuming the linear dynamics and Gaussian measurement [6, 31, 114]. There have been considerable efforts to generalize JPDA and overcome its shortcomings such as the extended Kalman Filter (EKF) to linearize modest nonlinear systems [47, 58, 68, 99, 110].

3.4.4 Particle Filtering

As we saw in Sec. 3.4.3, $P(f_k|\ Z_k)$ can be solved in closed form in (3.33) for the Gaussian linear model. For non-linear and/or non-Gaussian cases, the analytic solution of $P(f_k|\ Z_k)$ is usually very complex and impossible to solve.

Conditional density propagation, known as the *condensation algorithm* [13] and *Particle Filtering* [3, 42, 43], are *Sequential Monte Carlo* (SMC) algorithms, a *Monte Carlo* approximation of recursive Bayesian estimation (3.33). SMC methods have been widely used in problems where the Gaussian linear model is not satisfied. In such cases where computing an analytical solution for $P(f_k|\ Z_k)$ is not feasible, an approximation of $P(f_k|\ Z_k)$ is computed instead. To do this, SMC first computes the state *a priori density* $P(f_k|\ Z_{k-1})$, then *a posteriori density* is evaluated incorporating z_k, the measurement set at time k. SMC uses *importance sampling* to approximate $P(f_k|\ Z_k)$ as follows.

Importance Sampling

Importance sampling or factor sampling [13, 53] computes an approximation of an unknown density function, here $P(f_k|\ Z_k)$, by sampling from a known density assumed as *a priori density* $\tilde{P}(f)$. At first a set of N samples $S_f = \{s_f^1, s_f^2, \cdots, s_f^N\}$ representing system state variable f are drawn from $\tilde{P}(f)$ randomly. Then sample weighings $\{w^i|\ i \in [1, N]\}$, associated to samples $\{s_f^i|\ i \in [1, N]\}$, are computed using

$$W = \left\{ w^i|\ w^i = \frac{P(z|\ s_f^i)}{\sum_{n\in[1,N]} P(z|\ s_f^n)},\ i \in [1, N] \right\} \tag{3.39}$$

where $P(z|\ s_f^i)$ is the evaluation of $P(z|f)$ at sample s_f^i. A new sample set

$$S_{f|z} = \{s_{f|z}^1, s_{f|z}^2, \cdots, s_{f|z}^N\} \tag{3.40}$$

is then generated by sampling from sample set S_f (*a priori density*) applying the probability set W (measurement density).

To obtain the *a priori density* at each time step k, the system dynamics will be applied to sample set $S_{f|z}$ which represents the *a posteriori density* from time $k-1$. Then to obtain the *a posteriori density*, the prior density will be sampled with probability set W representing measurement density. As N increases, the distribution of $S_{f|\ z}$ increasingly tends to that of $P(f|z)$. Each sample weight w^i is known as importance weights and each pair of $(s^i_{f|z}, w^i)$ is called a particle or a weighted sample.

3.4.5 Multi-Hypotheses Tracking

Multi-hypotheses tracking (MHT) was originally developed by Reid [105] for multi-target tracking. In MHT, multiple hypotheses, including some likely and some unlikely hypotheses, are propagated over time.

As we discussed in Sec. 3.7, recursive Bayesian estimation sequentially approximates the system state f_k

$$\{f^h_k|\ h = 1, 2, ...\} \tag{3.41}$$

and searches in the hypothesis space for the best member

$$\hat{f}_k = f^{\hat{h}}_k \ \text{ where } \ \hat{h} = arg\ \left\{ \underset{h}{max}\ \ P(f^h_k|Z_{1:k}) \right\} \tag{3.42}$$

as the solution. In contrast with the recursive Bayesian estimation that associates only the measurement set at time k to the targets, MHT approximates the system state $F_{1:K}$ by association of the measurements over multiple frames $[1:K]$.

MHT is a Bayesian approach in which the likelihood of each hypothesis is computed by a probability function over all existing hypotheses. In each time step k similar hypotheses will be combined, unlikely ones will be deleted, and likely hypotheses will be retained. Therefore by deleting the unlikely hypotheses in each time step, MHT solves the tracking

problem by searching in a reduced hypothesis space. In this way, to solve

$$\hat{F}_{1:K} = arg \left\{ \underset{F_{1:K}}{max} \; P(F_{1:K}|\; Z_{1:K}) \right\} \tag{3.43}$$

MHT associates a sequence of measurement sets over $[1:K]$ to the targets by finding the most likely hypothesis from a limited hypothesis set over $[1:K]$

$$\{F^h_{1:K}|\; h = 1, 2, ...\} \tag{3.44}$$

thus finding the best member

$$\hat{F}_{1:K} = F^{\hat{h}}_{1:K} \;\; \text{where} \;\; \hat{h} = arg \left\{ \underset{h}{max} \; P(F^h_{1:K}|Z_{1:K}) \right\} \tag{3.45}$$

as the solution. Attributes of MHT can be summarized as

- Initiating a new track where new feature is observed,
- Retaining a track as long as feature to be associated to the track is observed,
- MHT terminates a track where feature to be associated to the track is not observed,
- A measurement can be associated at most to one target,
- At most one measurement can be associated to a target.

To estimate the present state considering all the previous states and measurement sets, the computational cost of MHT algorithm grows exponentially over time. Hence to manage the complexity, the present state is usually estimated over past N frames.

Chapter 4

The Proposed Approach

One of the most important and common tasks for biomedical researchers is cell tracking, however it continues to be undertaken manually. Researchers visually perform cell motion analysis and observe cell movement or changes in cell shape for hours to discover when, where and how fast it moves, splits or dies. This task is tedious and painful due to the often corrupted or blurred images, the presence of clutter, fixing eyes for a long time, and repeating the same task for different cell types. Furthermore, with the extent of cell imaging data ever increasing, manual tracking becomes progressively impractical. As a result, automated cell tracking systems are mandatory to further advance the study of biological cells [8, 17, 33, 37, 54, 90, 96, 128]. An automated tracking system performs automatic object tracking usually under challenging conditions, which also presents a very attractive and difficult research problem in computer vision and digital image processing.

4.1 The Problem at Hand

There are many applications of cell segmentation/localization, detection/recognition, and tracking in image sequences. Therefore, an automatic or semi-automatic tracking system has an immense need among biology and biomedical researchers. By eliminating the tedious process of manual cell tracing, such a system could be used to classify cells based on some quantitative and qualitative biological criteria.

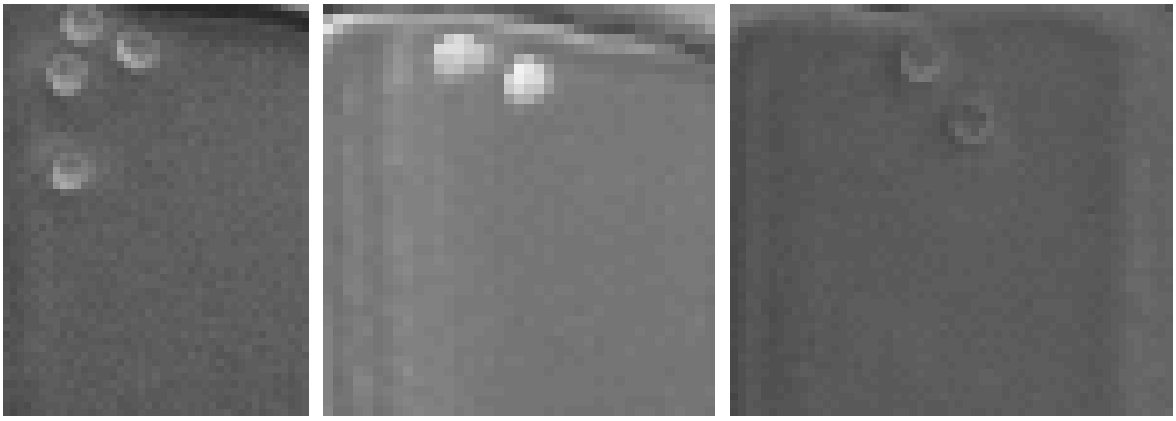
Due to the large number of cell types with different features and behaviours, designing a universal cell tracking system is impractical. In this research, we focus on Hematopoietic Stem Cells (HSCs), which proliferate and differentiate to different blood cell types continuously during their lifetime, and are of substantial interest in gene therapy, cancer, and stem-cell research. Fig. 4.1 shows three different HSC phenotypes.

The goal of this research is to perform living cell localization/segmentation and tracking. The proposed method is a Bayesian approach for multi-target tracking and association over sequences of phase contrast microscope images. This research has been performed in close cooperation with the Chemical Engineering Department of University of Waterloo and the Terry Fox Cancer Research Laboratory at British Columbia.

Depending on the nature of the tracking problem it may be categorized as one of i) segmentation-based, ii) detection-based, iii) recognition-based, and iv) localization-based tracking. Tracking HSCs over time is one of the most challenging tracking problems that belongs to the fourth tracking category, localization-based tracking.

4.2 Challenges and Difficulties

Available image sequences are taken over time from different cell types. Data sets include multiple long image sequences, showing the dynamic change of cells over time. The most



Phenotype 1 Phenotype 2 Phenotype 3

Figure 4.1: Different HSC phenotypes: (Phenotype 1) Cells have bright boundary and dark interior. (Phenotype 2) Cells are completely bright. (Phenotype 3) Poor contrast cells.

challenging task for an image-sequence tracking problem is to adapt and extend the available image processing approaches to the applications of cell imaging to analyze these data sets.

Moreover, living cell tracking as a localization-based tracking problem is a very challenging task. Living cell images generally are corrupted with noise, have a short range of gray levels as opposed to most segmentation/tracking problems which focus on known objects in high contrast data sets such as video images and movies. Although cell staining techniques can be used to increase the contrast between cell areas and background, these techniques undesirably stain different parts of a tissue unevenly, causing inhomogeneity. Further, the presence of many cluttered objects, insignificant texture, and weak cell boundaries are the other reasons which make cell tracking more challenging.

In comparison with tracking problems which track known objects in high contrast video clips, cell tracking is a relatively new application in which standard image tracking algorithms, regarding the differences between two successive frames, usually would yield

meaningless results in the cell-imaging context.

The key challenges of our tracking problem can be summarized as:

1. To keep cells alive and healthy, light exposure must be controlled during their life cycle to minimize phototoxicity. Therefore it is desired to limit light exposure in each frame and to sample the frames as far apart as possible, leading to infrequent, poorly-contrasted images, directly at odds with the data desired for easy tracking: frequent, high-contrast images.

2. The limited light exposure and cell transparency both contribute to the very low contrast images that make the localization/segmentation task very difficult.

3. The blood stem cells which are of interest in this research, all have the same visual appearance and cannot be discriminated visually.

4. To track a particular cell over time, the association task becomes crucial.

4.3 Ideas and Directions

In this research, a model-based cell tracker is designed to locate and track individual cells. The proposed cell tracking system consists of three inter-related stages:

- Cell detection/localization,
- The association of detected cells,
- Background estimation/subtraction.

The key ideas and directions of this research can be summarized as

- Ridgelet BayesShrink for image denoising by employing Ridgelet transform for abstract representation of lines,
- The design of a model-based object localization for HSCs of Phenotype 1,
- The design of a general model-based object localization for different HSC phenotypes investigated in this research,
- Present object localization as an inverse problem and provide two deconvolution based approaches,
- Develop a mutual method for foreground localization and background estimation,
- The introduction of an optimal generative single-frame association for multi-target tracking based on an extended Hungarian method which is a class of linear programming optimization,

The proposed ideas and directions will be addressed in detail in the rest of this thesis as following:

- *Denoising (Chapter 5)*
- *Cell Segmentation and Localization (Chapter 6)*
- *Background Estimation (Chapter 7)*
- *Cell Tracking (Chapter 8)*

Chapter 5

Denoising

Recently the Ridgelet and Curvelet transforms were developed to reduce the limitations of wavelet-based image processing. The wavelet transform effectively represents the point singularities of one-dimensional signals, however it produces several large coefficients along significant edges of an image even at fine scales. Hence, edges of an image appear as many large wavelet coefficients repeatedly at fine scales, so to properly reconstruct the edges of the image many wavelet coefficients are required. The estimation of so many coefficients makes wavelet denoising techniques complex. The ridgelet transform can compensate the weaknesses of the wavelet transform to represent smoothness along the edges of 2-D images. In this chapter, the ridgelet transform is discussed and our proposed ridgelet based image denoising is introduced. The following concepts are addressed in this chapter.

- *Ridgelet Transform*
- *Ridgelet Denoising*
- *Calculating BayesShrink Thresholds for Ridgelet Denoising*
- *Combined Ridgelet-Wavelet Denoising*

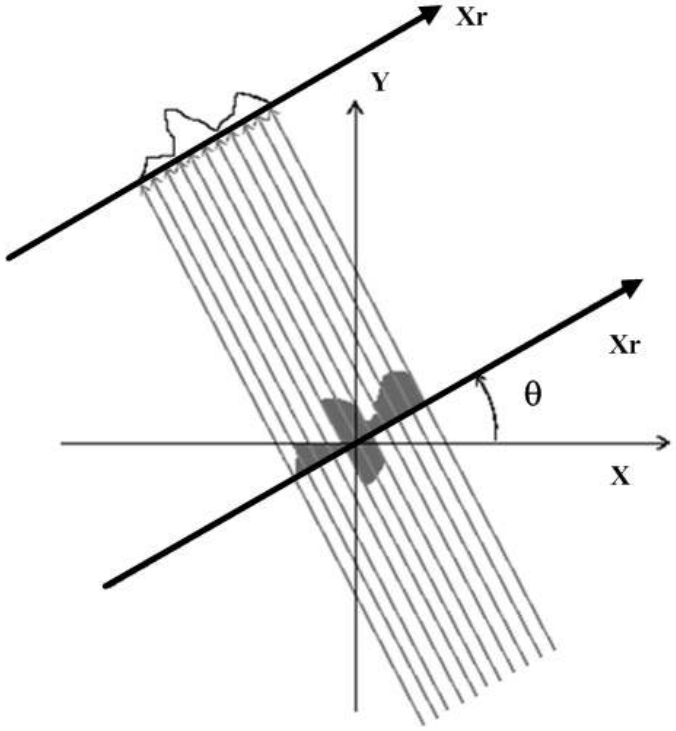
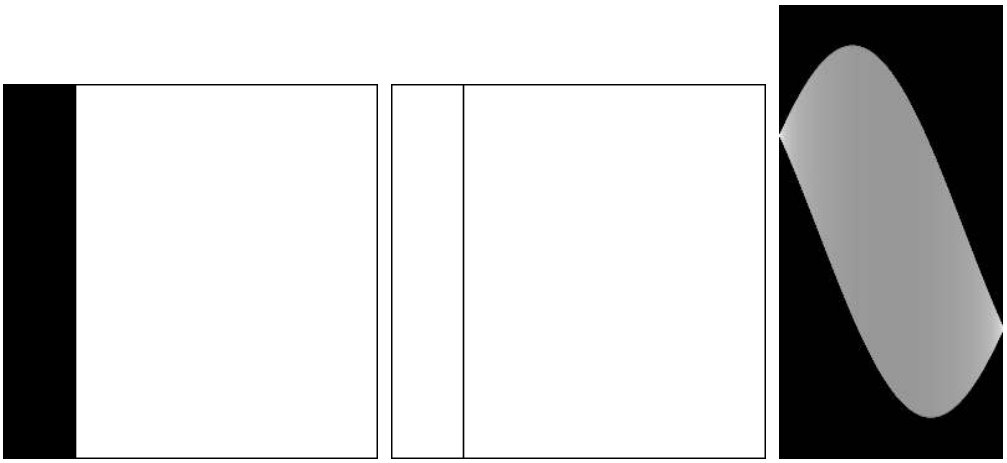


Figure 5.1: Radon transform of a 2-D surface computed for direction θ.

5.1 Ridgelet for Image Denoising

An edge, as the border of two smooth regions, is generally a smooth curve, however it is discontinuous and can be considered as a 1-D singularity in a piecewise 2-D image. The wavelet transform faces some difficulties to discover edges as 1-D singularities in 2-D signals. The 2-D wavelet transform as the product of 1-D wavelets, discovers the singularities across the edge but it doesn't recognize the smoothness along the edge. To compensate for this weakness of the wavelet transform in higher dimensions, ridgelet and curvelet transforms were recently introduced by Candes and Donoho [18, 19]. Different denoising methods have been proposed for signal denoising via wavelet. On the other hand



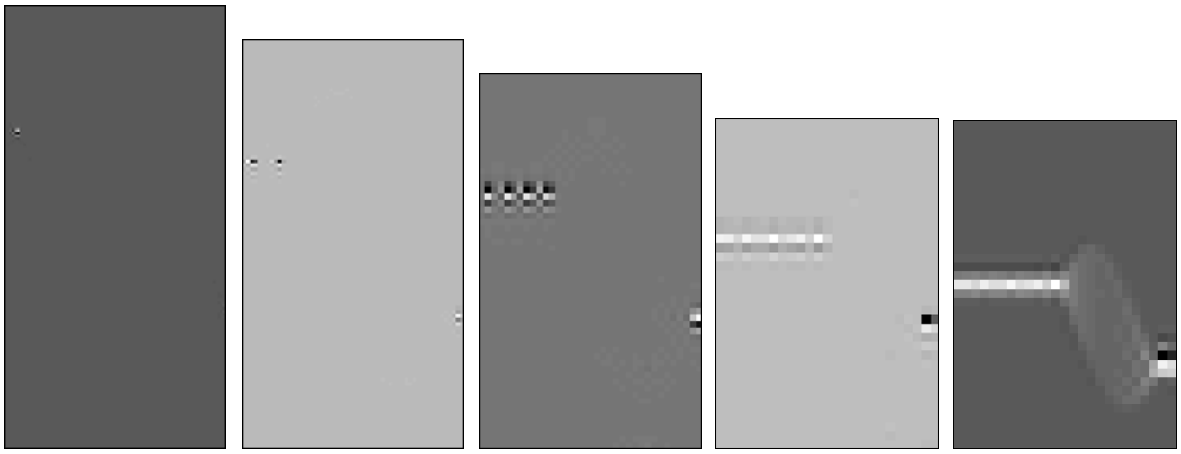
(a) (b) (c)

Figure 5.2: (a) A synthetic image consisting of white and black regions. (b) Edge image obtained applying *Sobel* kernel to (a) for edge detection. (c) Radon transform of edge image (b).

VisuShrink ridgelet thresholding has been recently introduced [39] as an alternative to the wavelet denoising and performs better than wavelet for images with straight lines. In this chapter our *BayesShrink* ridgelet image denoising is introduced and the results are compared with those of the *VisuShrink* ridgelet method.

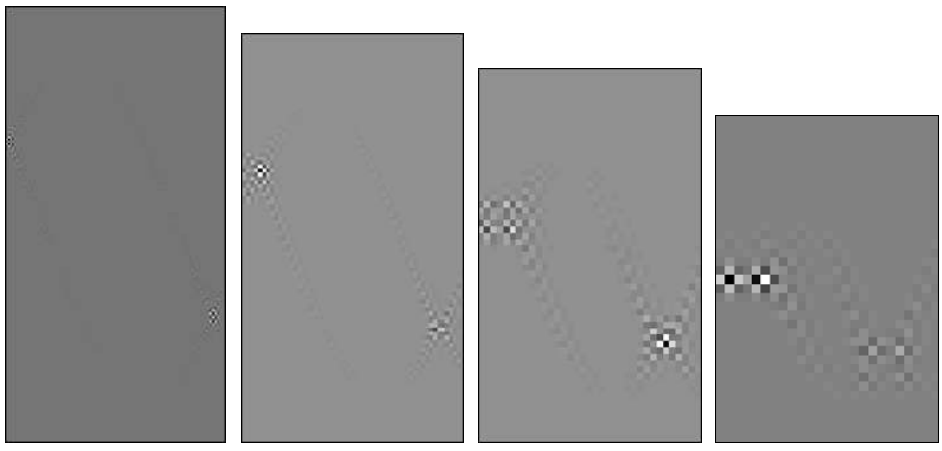
5.1.1 Ridgelet Transform

The ridgelet transform effectively represents line singularities of 2-D signals. Since a sparse representation of smooth functions and straight edges is provided by the ridgelet transform, this new expansion can accurately represent both smooth functions and edges with fewer nonzero coefficients and achieves a lower mean square error (MSE) than the wavelet transform. It maps the line singularities into point singularities in the Radon domain



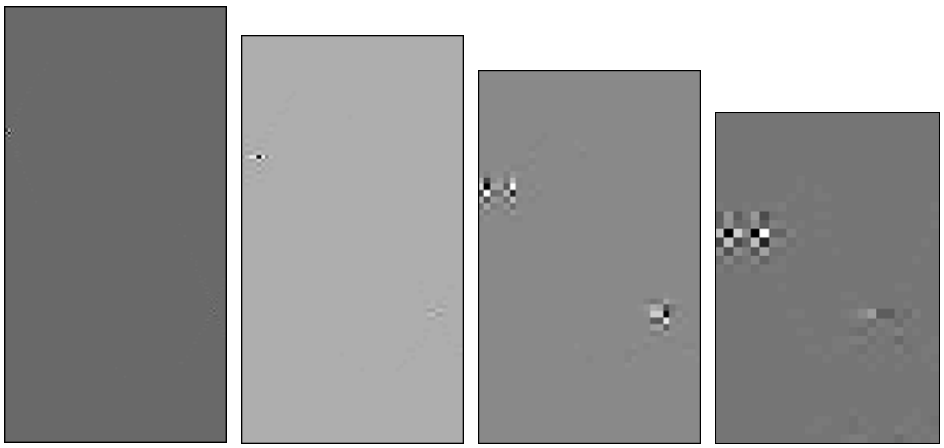
(a) (b) (c) (d) (e)

Figure 5.3: Ridgelet horizontal and the approximate coefficients of the image depicted in the Fig. 5.2(b): (a) Level 1. (b) Level 2. (c) Level 3. (d) Level 4. (e) Approximate coefficients.



(a) (b) (c) (d)

Figure 5.4: Ridgelet vertical coefficients of the image depicted in the Fig. 5.2(b): (a) Level 1. (b) Level 2. (c) Level 3. (d) Level 4.



(a) (b) (c) (d)

Figure 5.5: Ridgelet diagonal coefficients of the image depicted in the Fig. 5.2(b): (a) Level 1. (b) Level 2. (c) Level 3. (d) Level 4.

[18] by employing the embedded Radon transform. Therefore, the wavelet transform can efficiently be applied to discover the point singularities in this new domain.

The ridgelet transform can be considered as application of wavelet transform in the radon domain. As we can observe in Fig. 5.1, the radon transform represents a two dimensional function by a set of line integrals in (X_r, Y_r) axis that have angular direction θ with respect to (X, Y) axis. In this way a line can be represented with a few significant radon coefficient as it is shown in Fig. 5.2. Because of the abstract representation of the lines in radon domain, the application of the wavelet transform to the radon coefficients produces a few large coefficients along significant edges. As a result to properly reconstruct the edges of the image, only few wavelet coefficients are required. Needing to estimate few coefficients makes the denoising task much less complex. The detailed coefficients including horizontal, vertical, and diagonal for the edge image (depicted in Fig. 5.2) are depicted in Fig. 5.3, 5.4, and 5.5 respectively, while approximate ridgelet coefficients is shown in Fig.

5.3(e).

Having the ability to approximate singularities along a line, several terms with common ridge lines can effectively be superposed by the ridgelet transform. The bivariate ridgelet transform [18] in R^2 is defined by

$$\Re_{\alpha,\beta,\theta}(\kappa) = \alpha^{-1/2}\omega((\kappa_1 \cos\theta + \kappa_2 \sin\theta - \beta)/\alpha) \tag{5.1}$$

where, $\alpha > 0, \beta$ and θ are scale, location and orientation parameters respectively and ω is a univariate wavelet function on $R \to R$. Along the ridgelet lines $\kappa_1 \cos\theta + \kappa_2 \sin\theta$, ridgelets are constant and they are equal to the wavelets in the orthogonal direction. Ridgelet coefficients of a bivariate function $I(\kappa)$ in R^2 are given by

$$\Re_I(\alpha, \beta, \theta) = \int \Re_{\alpha,\beta,\theta}(\kappa) I(\kappa) d\kappa \tag{5.2}$$

The reconstruction formula is given by

$$I(\kappa) \;=\; \int_0^{2\pi} \int_{-\infty}^{+\infty} \int_0^{\infty} \Re_I(\alpha, \beta, \theta) \Re_{\alpha,\beta,\theta}(\kappa) \frac{d\alpha}{\alpha^3} d\beta \frac{d\theta}{4\pi} \tag{5.3}$$

and is valid for integrable (and square integrable) functions. Considering the 2-D ridgelet transform as a 1-D wavelet transform in the radon domain, the ridgelet coefficients of function $I(\kappa)$ can be defined as

$$\Re_I(\alpha, \beta, \theta) = \int \Re_t(\theta, \tau) \alpha^{-1/2} \omega((\tau - \beta)\alpha) d\tau \tag{5.4}$$

where $\Re_t(\theta, \tau)$ is the Radon transform of function $I(\kappa)$.

5.1.2 Ridgelet Denoising Concept

The wavelet denoising was explained in chapter 3 (Sec. 3.1), while ridgelet denoising will be discussed in this chapter. To explain the ridgelet denoising procedure, let S be an original $M \times M$ image, where $i = 1, 2, \cdots, M$, $j = 1, 2, \cdots, M$, and $I[i, j] = S[i, j] + n[i, j]$ is

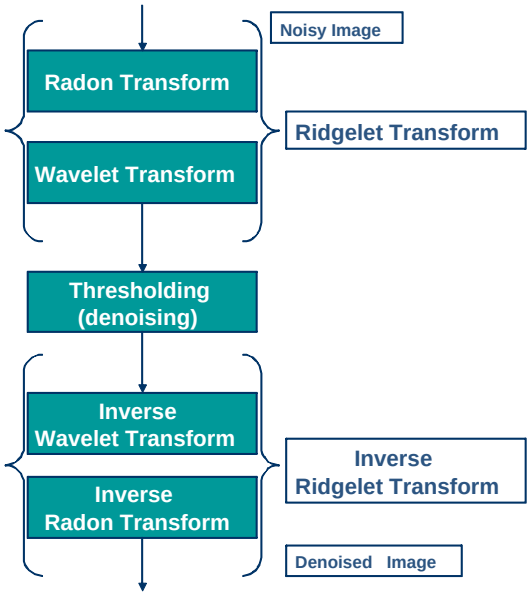


Figure 5.6: Ridgelet for image denoising

the image corrupted by additive noise n which is identically distributed and independent of S. As depicted in Fig. 5.6, in the first step of ridgelet denoising, the observed image I is transformed into the ridgelet domain. Then the ridgelet coefficients are thresholded and finally the denoised coefficients are transformed back to reconstruct the image. Let R_D and R_R be the forward ridgelet decomposition and inverse ridgelet reconstruction, respectively. Let T and τ be the thresholding operator and the threshold value respectively. The ridgelet thresholding can be summarized as

$$\begin{aligned} I_R &= R_D(I) \\ I_\tau &= T(I_R, \tau) \\ \hat{S} &= R_R(I_\tau) \end{aligned} \tag{5.5}$$

The choice of the threshold and the method which is used to calculate the threshold, determine how efficient the denoising technique would be. Although selecting a small threshold may produce an output image close to the input, the recovered image may still be noisy. On the other hand, a choice of a large threshold may yield a blurred image by setting most of the wavelet coefficients to zero. Two different thresholding techniques, VisuShrink and BayesShrink, are explained in the following Section.

5.1.3 BayesShrink Method

The subband wavelet and ridgelet coefficients of a natural image can be described by the Generalized Gaussian Distribution (GGD) [22, 23]:

$$GG_{\sigma,I_R,\gamma}(I_R) = P(\sigma_{I_R},\gamma)\exp\{-[\delta(\sigma_{I_R},\gamma)\ |I_R|\ ]^{\gamma}\} \tag{5.6}$$

where $-\infty < i_R \in I_R < +\infty, \gamma > 0$ and,

$$\delta(\sigma_{I_R},\gamma) = \sigma_{I_R}^{-1}\left[\frac{\Gamma(3/\gamma)}{\Gamma(1/\gamma)}\right]^{\frac{1}{2}} \tag{5.7}$$

and,

$$P(\sigma_{I_R},\gamma) = \frac{\gamma\cdot\delta(\sigma_{I_R},\gamma)}{2\Gamma(\frac{1}{\gamma})} \tag{5.8}$$

σ_{I_R} is the standard deviation of subband ridgelet coefficients, γ is the shape parameter and Γ is Gamma function. Considering such a distribution for the ridgelet coefficients and estimating γ and σ_{I_R} for each subband, the soft threshold τ which minimizes the Bayesian Risk [22, 23] can be obtained by minimizing

$$\Re(\tau) = E(\hat{I}_R - I_R)^2 = E_{I_R}E_{J|I_R}(\hat{I}_R - I_R)^2 \tag{5.9}$$

where $\hat{I}_R$ is $\tau(J)$, $J|I_R$ is $N(I_R,\sigma)$ and I_R is $GG_{\sigma_{I_R},\gamma}$. Then the optimal threshold τ^* is given by

$$\tau^*(\sigma_{I_R},\gamma) = arg\ \min_{\tau}\ \Re(\tau) \tag{5.10}$$

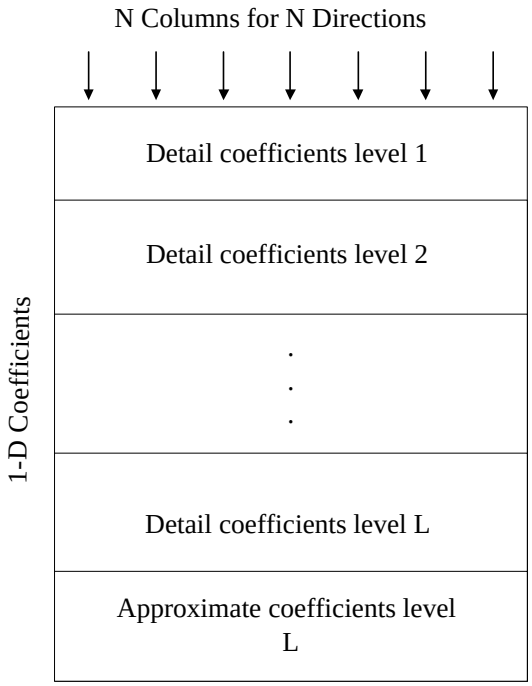


Figure 5.7: Ridgelet coefficients of a 1-D signal after L decomposition levels.

τ^* does not have a closed form solution and numerical calculation is used to find τ^*. An estimation of the value τ^* is concluded by setting the threshold to

$$\hat{\tau}(\hat{\sigma}_{I_R}) = \frac{\hat{\sigma}_n}{\hat{\sigma}_{S_R}} \tag{5.11}$$

where $\sigma^2_{S_R}$ is variance of subband ridgelet coefficients and σ^2_n is an estimation of the noise variance.

5.1.4 Calculating the BayesShrink Threshold for Ridgelet Coefficients

Subband dependent thresholds are used to calculate the BayesShrink ridgelet threshold. The estimated threshold is given by (5.11). The 1-D ridgelet coefficients, obtained by com-

puting the wavelet transform of 1-D radon coefficients, corresponding to different directions are depicted in Fig. 5.7. In this figure each column corresponds to a specific direction, hence the number of columns determines the number of directions and each column contains subband detail coefficients for L different decomposition levels. To estimate the noise variance σ_n^2 from the subband details, the median estimator is used on the 1-D subband coefficients:

$$\hat{\sigma}_n = median(|Details|)/\gamma \tag{5.12}$$

where γ assumed to be equal to 0.6745 as it has been used in [41]. The signal standard deviation is calculated for each direction in each subband detail individually. Thus having N directions and L subbands, $N \times L$ different σ_{I_R} must be estimated corresponding to $N \times L$ subband-directions coefficients. Note that in BayesShrink wavelet denoising, σ_{I_w} is estimated on 2-D dyadic subbands [22]. Thus having L decomposition levels, $3 \times L$ different σ_{S_R}, signal ridgelet subband standard deviation, must be estimated to calculate the thresholds for the different subbands. To estimate σ_{S_R}, recall that the observed signal I considered to be $I = S + n$, so for the ridgelet coefficients we have $I_R = S_R + n$, where signal ridgelet coefficients (S_R) and noise (n) are assumed to be independent. Therefore, $\sigma_{I_R}^2 = \sigma_{S_R}^2 + \sigma_n^2$ where $\sigma_{I_R}^2$ is the variance of the observed signal ridgelet coefficients. So $\hat{\sigma}_{S_R}$ is estimated by

$$\hat{\sigma}_{S_R} = \sqrt{max((\hat{\sigma}_{I_R}^2 - \hat{\sigma}_n^2), 0)} \tag{5.13}$$

5.2 Combined Ridgelet-Wavelet Approach

As we can observe in the depicted figures in the results section, the ridgelet performs better than wavelet for denoising edges and lines in a given image, while wavelet overall

Algorithm 7 Combined Wavelet-Ridgelet Denoising Algorithm

1: $e = \alpha\hat{\sigma}_n^2$

2: $\hat{S} = I$

3: $C_w = \text{Wavelet Transform}\{\hat{S}\}$

4: $C_{T_w} = \text{BayesShrink Wavelet}\{C_w\}$

5: $\hat{S} = \text{Inverse Wavelet Transform}\{C_{T_w}\}$

6: $C_r = \text{Ridgelet Transform}\{\hat{S}\}$

7: $C_{T_r} = \text{BayesShrink Ridgelet}\{C_r\}$

8: $\hat{S} = \text{Inverse Ridgelet Transform}\{C_{T_r}\}$

performs better, specifically on natural images. Therefore, it seems quite reasonable to take advantage of the ridgelet transform for improving the performance of the wavelet transform to denoise edges and lines. Hence, a combined denoising method is proposed in this section as depicted in Fig. 5.8. In the proposed method, the overall better performance of the wavelet transform for image denoising and the better performance of ridgelet transform for denoising edges are mutually gained to improve the visual image quality. Our proposed method by combining wavelet and ridgelet, performs better than each method individually. The proposed denoising method follows.

5.2.1 Combined Denoising Algorithm

As we can observe in the denoised image depicted in Fig. 5.16(Top) using BayesShrink wavelet denoising method, imperfect noise removal causes some artifacts especially in vicinity of edges and lines in the image. As it was discussed in this chapter, these artifacts are the effect of incapability of the wavelet transform to discover the smoothness along the edges and lines, representing them with many significant coefficients that makes the denoising

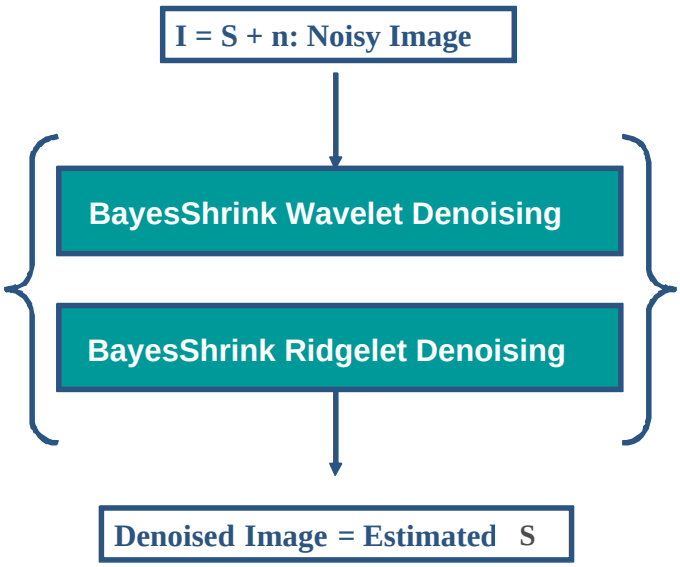


Figure 5.8: Combined denoising method

task very complex along these 1-D singularities.

Therefore, to improve the denoising along edges and lines, we take advantage of ridgelet transform that effectively discover the smoothness along edges by representing them using few significant coefficients. To do this, the observed image will be passed through wavelet and ridgelet denoising filters sequentially in the proposed method as depicted in Algorithm 7.

5.2.2 The Next Step as Potential Extension

The proposed combined denoising method potentially can be extended as an iterative method as depicted in Fig. 5.9. To avoid the overhead complexity that signal synthesizing methods such as *Basis Pursuit* and *Matching Pursuit* [30, 89] might cause, in the iterative method the observed image may iteratively passe through wavelet and ridgelet denoising filters until a cost function is satisfied as depicted in Algorithm 8. Assuming $I = S + n$, the

Algorithm 8 Iterative Combined Wavelet-Ridgelet Denoising Algorithm

$e = \alpha\hat{\sigma}_n^2$
$\hat{S}_t = I$
repeat
  $\hat{S}_{t-1} = \hat{S}_t$
  $C_w =$ Wavelet Transform$\{\hat{S}_t\}$
  $C_{T_w} =$ BayesShrink Wavelet$\{C_w\}$
  $\hat{S}_t =$ Inverse Wavelet Transform$\{C_{T_w}\}$
  $C_r =$ Ridgelet Transform$\{\hat{S}_t\}$
  $C_{T_r} =$ BayesShrink Ridgelet$\{C_r\}$
  $\hat{S}_t =$ Inverse Ridgelet Transform$\{C_{T_r}\}$
until $|\hat{S}_t - \hat{S}_{t-1}|^2 < e$

noise variance can be estimated and the denoising task will be repeated until $|\hat{S}_t - \hat{S}_{t-1}|^2 < e$ be satisfied, where $\hat{S}_t$ and $\hat{S}_{t-1}$ are estimated pure signal at present and previous time steps and e can be selected as a fraction of the noise variance.

5.2.3 Results

In this section the proposed ridgelet denoising and combined wavelet-ridgelet denoising techniques are used to recover the noisy images which are corrupted with additive white noise.

BayesShrink Ridgelet Denoising vs. VisuShrink Ridgelet Denoising

BayesShrink and VisuShrink ridgelet image denoising methods are implemented and based on different wavelet basis the results are compared. Since the ridgelet transform performs

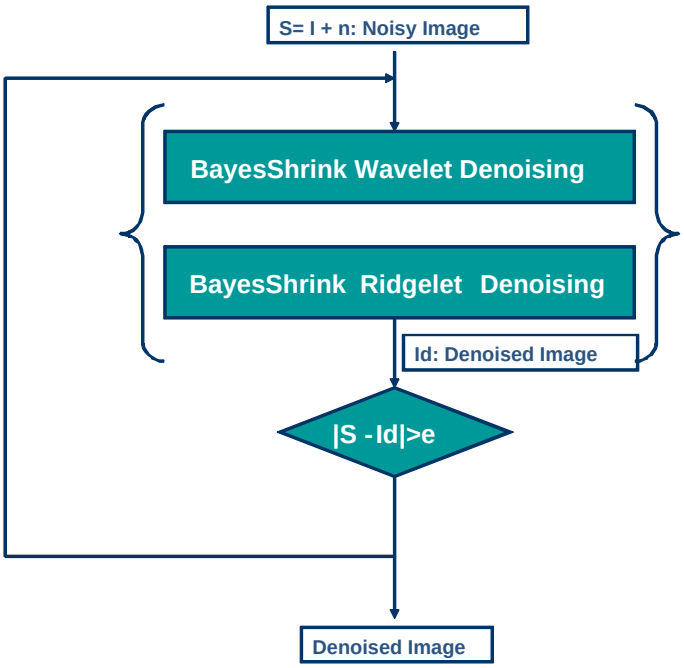
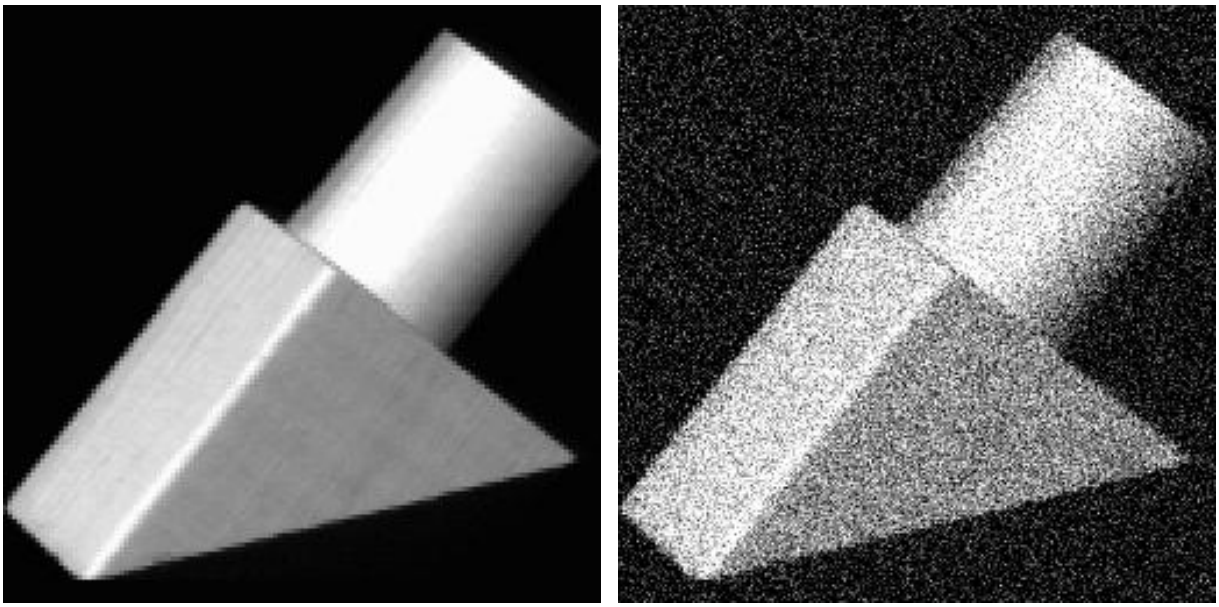


Figure 5.9: Iterative combined denoising method

better on images having many edges and lines, the test image in the following experiments, as depicted in Fig. 5.10, is an image with perfectly straight lines which has been used in [39]. Denoised images depicted in Fig. 5.11 and Fig. 5.12 are derived using the VisuShrink and BayesShrink thresholding methods respectively. The results are obtained based on three different wavelet bases including Daubechies, Symlets and Biorthogonal. As we can observe according to the SNR measurements, the results obtained by BayesShrink ridgelet method are better than those obtained by VisuShrink ridgelet method using different wavelet bases. Moreover, BayesShrink provides superior results than VisuShrink based on image quality regardless of the wavelet bases.



(a) (b)

Figure 5.10: (a) Original image. (b) Noisy image with SNR = 7.22.

Combined BayesShrink Denoising vs. BayesShrink Wavelet Denoising

Some results obtained by applying BayesShrink wavelet, BayesShrink ridgelet and the proposed method are presented in this section. A synthetic image with straight lines is used in the first experiment, as shows in Fig. 5.13. As we can observe, the combined filtering performed better than the two others such that its results have better visual quality and higher SNR. Fig. 5.14 shows the denoised Lena image using BayesShrink wavelet, BayesShrink ridgelet and the proposed method. Although this image doesn't have straight regions, the combined BayesShrink wavelet-ridgelet does not degrade the superior wavelet result in this domain and performs as well as BayesShrink wavelet. Finally, the proposed method is applied to the Gold-hill image, a natural image with straight regions. Figs. 5.15(Top) and (Bottom) show the original and noisy image respectively. Restored images using BayesShrink wavelet and the proposed method are depicted in Fig. 5.16(Top) and (Bottom) respectively. As can be observed, not only the SNR is improved by the combined method but the image is smoother and free of the many local artifacts clearly visible in the BayesShrink wavelet case.

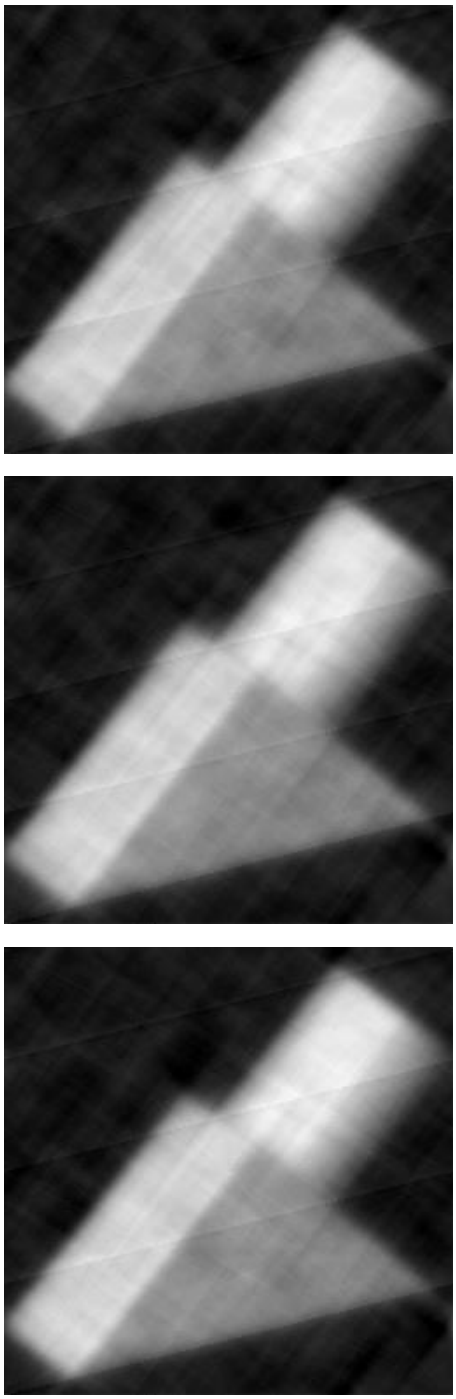


Figure 5.11: VisuShrink ridgelet image denoising: (Top) SNR = 11.56 using daubechies wavelet basis. (Middle) SNR = 11.65 using symlets wavelet basis. (Bottom) SNR = 12.04 using biorthogonal wavelet basis.

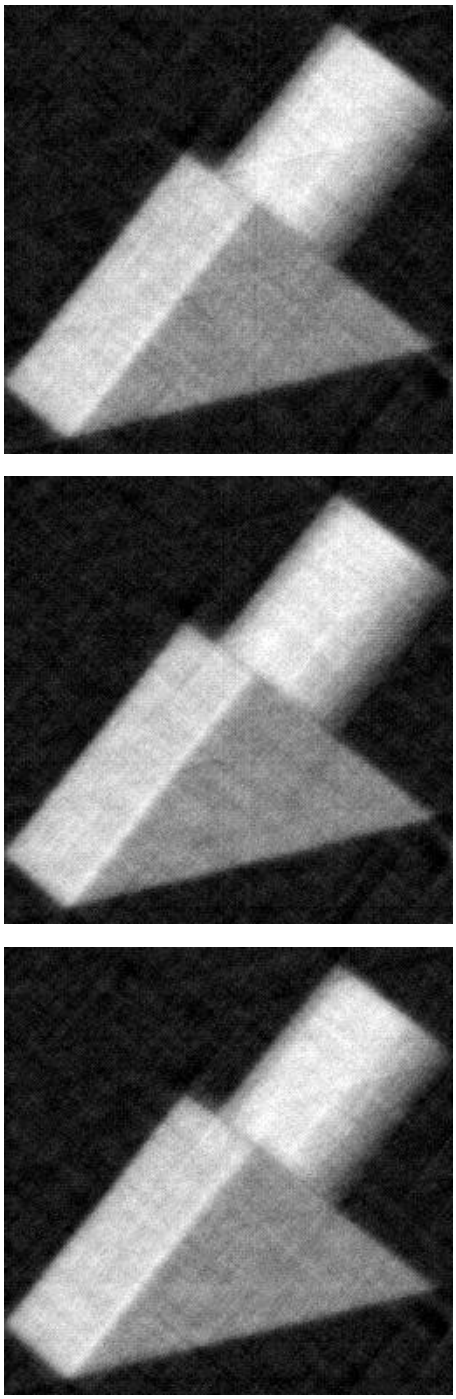


Figure 5.12: BayesShrink ridgelet image denoising: (Top) SNR = 13.25 using daubechies wavelet basis. (Middle) SNR = 13.30 using symlets wavelet basis. (Bottom) SNR = 13.16 using biorthogonal wavelet basis.

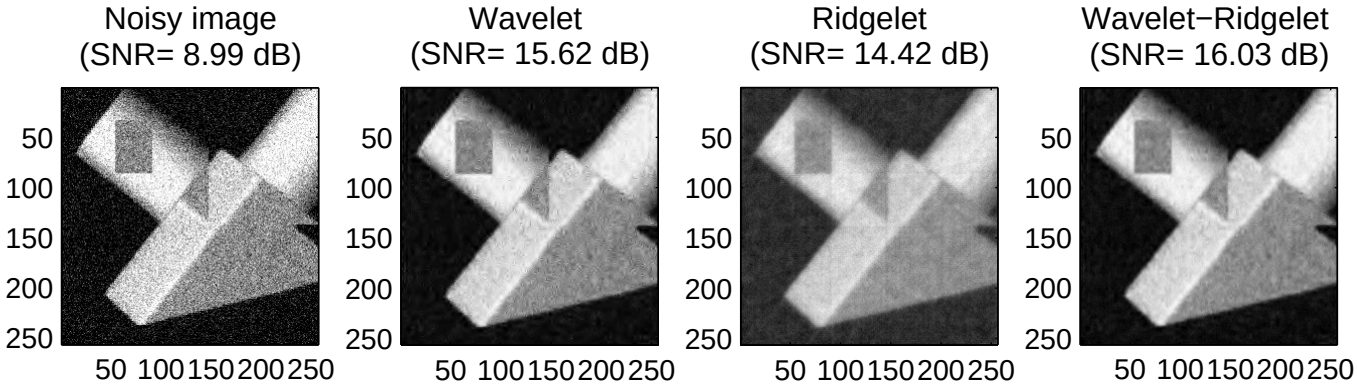


Figure 5.13: BayesShrink Methods to Restore a Noisy image

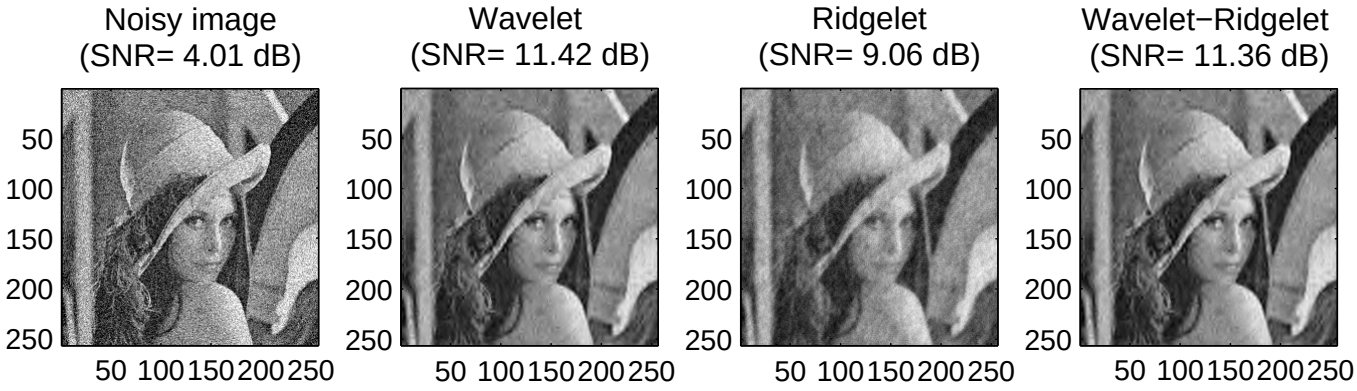


Figure 5.14: BayesShrink Methods to Restore Noisy Lena Image

Figure 5.15: (Top) Original Goldhill image. (Bottom) Noisy image with SNR=7.93.

Figure 5.16: (Top) BayesShrink Wavelet Denoising, SNR = 11.59. (Bottom) BayesShrink Wavelet-Ridgelet, SNR = 11.82.

Chapter 6

Cell Segmentation and Localization

As it was mentioned in chapter 4, based on the nature of tracking problem, they can be grouped in four categories: i) segmentation-based, ii) detection-based, iii) recognition-based, and iv) localization-based. Therefore segmentation, detection, recognition, and localization are crucial stages respectively.

The HSCs in our research have fairly circular shape and the same visual appearance, thus they cannot be discriminated visually and the tracking problem belongs to the localization-based tracking category. In turn, to track a particular cell over time, the association task becomes crucial.

To solve this localization-based tracking problem, HSCs must be first detected and/or segmented. A wide variety of methods have been proposed for cell segmentation [4, 8, 12, 17, 56, 90, 92, 127, 128] divided into six major categories: region based methods(Sec. 3.2.1), boundary based methods(Sec. 3.2.2), hybrid methods (Sec. 3.2.5), thresholding methods (Sec. 3.2.7), mean shift procedure (Sec. 3.2.8), and deformable models. Typical methods of these categories are split & merge [12], region growing methods [127], morphological operators [4], watershed [90], thresholding [56, 74, 128], mean shift procedure [33], nearest

neighbor graphs [54]and snakes [92].

To minimize phototoxicity and to keep cells alive and healthy, light exposure is limited during imaging HSCs that leads to poorly-contrasted images. AS a result, HSCs cannot be well segmented, rather they must be detected in each frame. The cell detection problem here, can be addressed as an anomaly detection problem, the localization of groups of pixels inconsistent with the random behavior of the image background. Moreover, HSCs in our study have fairly regular shape and brightness patterns. Hence, integration of these useful information in a detection method should improve the detection performance in comparison with simple thresholding methods. Our proposed segmentation and localization methods are addressed in this chapter as follows:

- *Problem Context (Sec. 6.1)*
- *Initial Cell Model (Sec. 6.2)*
- *Thresholding Cell Detection (Sec. 6.3)*
- *Probabilistic Cell Detection (Sec. 6.4)*
- *Deconvolution Model for Stem-Cell Localization (Sec. 6.5)*
- *Watershed-Deconvolution Method for Cell Detection (Sec. 6.6).*

6.1 Problem Context

Suppose we have an image sequence $I_{1:K} = \{I_1, I_2, ..., I_K\}$, the optimum answer to the detection problem in a typical frame I_k is the maximum a posteriori estimation of $z = (x_c, y_c, r)$

$$\hat{z}_k = arg \left\{ \max_{z_k} \; P(z_k|\; I_k) \right\} \tag{6.1}$$

A typical microscopic multi-well video clip in our experiments consists of 32 independent wells, in each of which two to four HSCs are injected. The blood stem cells have a fairly regular shape and brightness pattern which can be integrated in an initial cell model as follows in the next section.

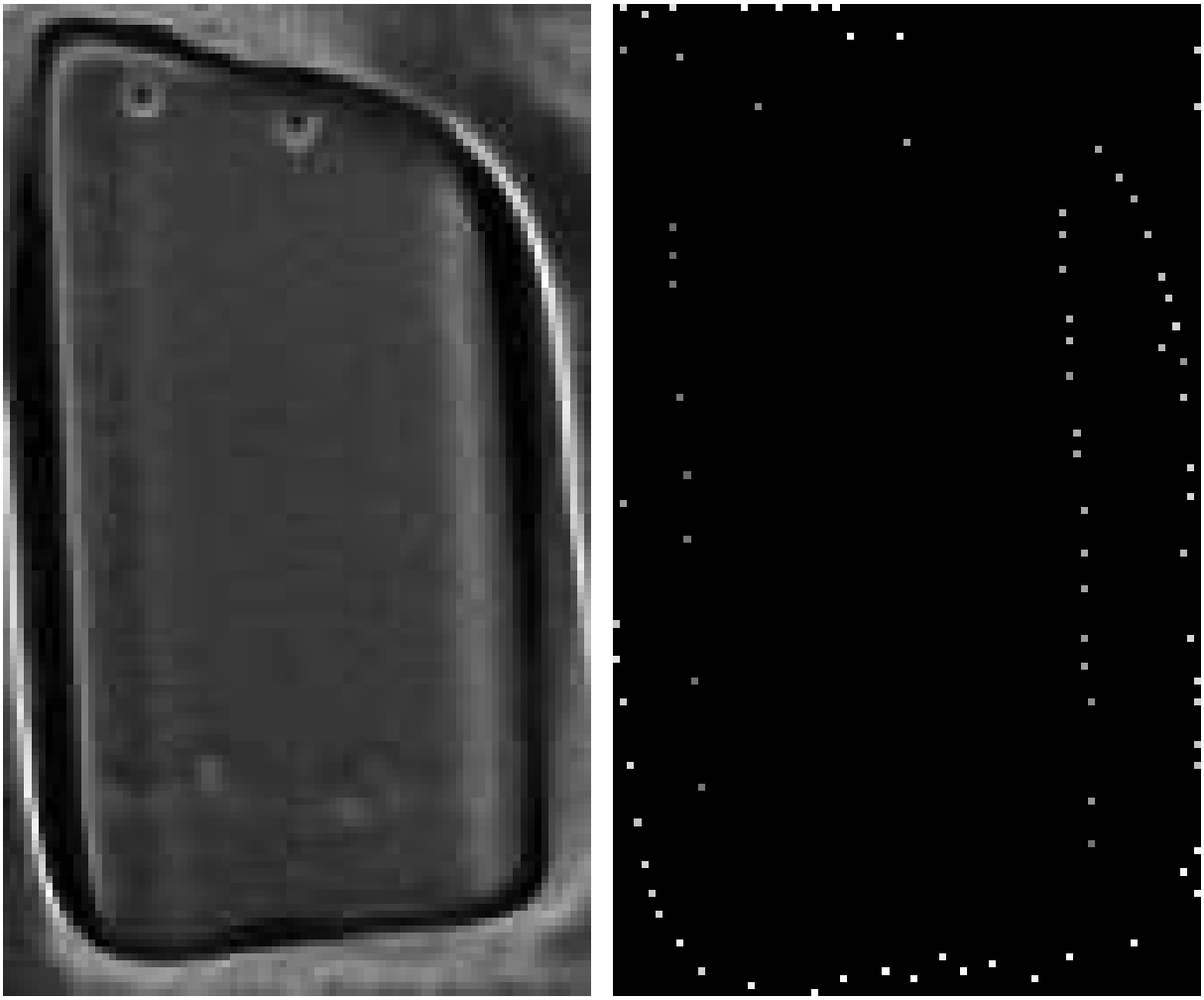
Single-well video clips are cropped from the original multi-well video clip and are processed individually. As we can observe in Fig. 6.1 complicated structures, such as well boundaries, are visible in coarsely cropped image sequences. These structures might be detected as cells and would degrade the cell detection performance. To avoid this, it will be assumed that wells are precisely cropped so that well structures are not visible in the image sequence, leading to a uniform background as depicted in Fig. 6.2. In the rest of this chapter we assume that the image sequences have a uniform background. How to achieve a uniform background is a question that will be answered in the chapter 7 where we discuss background estimation/subtraction problem.

6.2 Initial Cell Model

Looking at microscope images in Fig. 6.3, we can observe that HSCs can be characterized as an approximately circular object with a dark interior and a bright boundary — an effect due to phase contrast imaging modality. During splitting, a mature cell is divided to give birth to two new cells, as marked by a circle in Fig. 6.3(b).

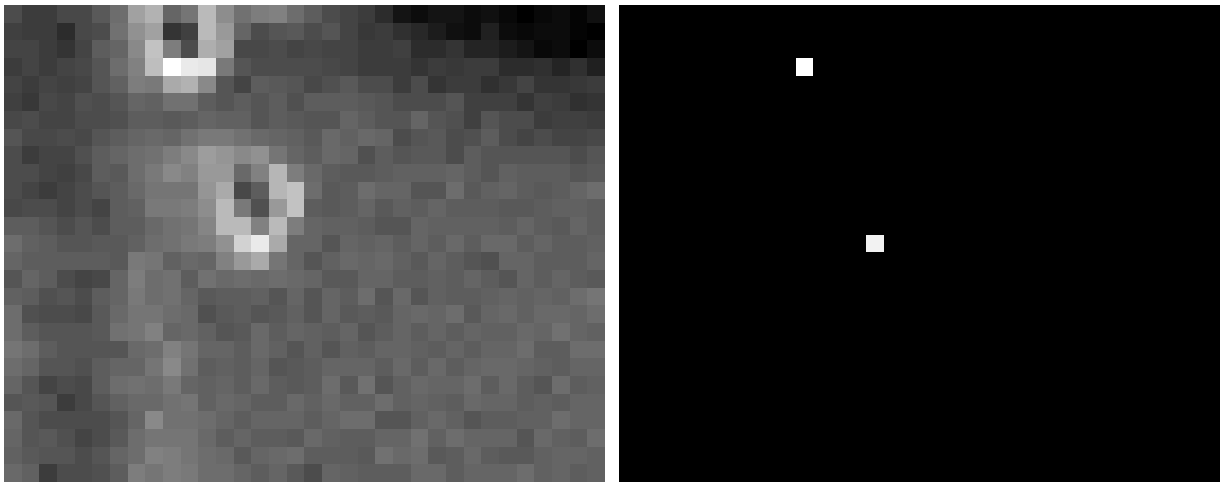
6.2.1 Feature Set

We can make the detection problem (6.1) much more specific by seeking particular features consistent with HSCs. So rather than a heuristic thresholding approach, the specific, consistent cell attributes observed should allow us to formulate a far more specific model,



(a) (b)

Figure 6.1: (a) Coarsely cropped well in which well boundaries are visible. (b) Applying the proposed model-based detection method (Sec. 6.2), to the coarsely cropped well. In contrast with the perfect result obtained by applying the proposed method to a cropped well interior with no boundaries (Fig. 6.2(b)), here the proposed method (Sec. 6.2) performs very poorly where the cropped image contains visible well boundaries.



(a) (b)

Figure 6.2: (a) Cropped well interior with no well boundaries. (b) Applying the proposed model-based detection method (Sec. 6.2), to the cropped well interior.

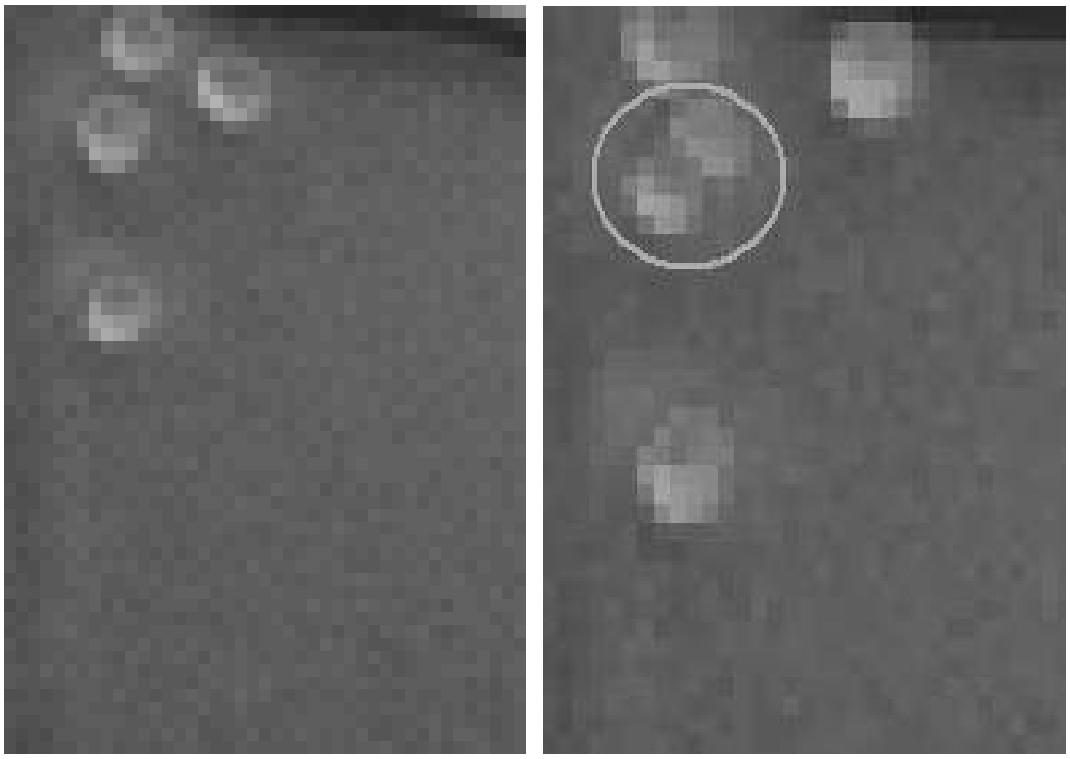
essentially a matched filter [123], to be more robust to noise and low contrast. The model may consider the following criteria:

- Cell size,
- Boundary brightness,
- Interior brightness,
- Boundary uniformity or symmetry.

The four cell criteria are then combined to formulate the following probabilistic cell model

$$P(x_c, y_c, r|\ I_k) = P_{cb}(\bar{B}) \cdot P_{ci}(\bar{C}) \cdot P_{bu}(B) \tag{6.2}$$

where the cell boundary P_{cb}, cell interior P_{ci} and boundary uniformity P_{bu} terms are elaborated below.



(a) (b)

Figure 6.3: HSC phase contrast image: (a) Non dividing cells. (b) A mature cell dividing into two new cells.

Boundary and Interior Cell Pixels

The proposed probabilistic cell model is designed to segment the individual cells within the same frame considering a cell as a dark region with a bright boundary. The most probable cell locations are identified in the image by integrating the probabilities of the average cell boundary intensity, the average inside cell intensity and uniformity of the cell boundary. As depicted in Fig. 6.4(b), to model a dark region surrounded by a bright boundary, the proposed cell model consists of two concentric circles, with the radius of the internal circle being half that of the external one. To facilitate the analysis of the image as a function of cell center location (x_c, y_c) and radius r we construct the set of boundary pixels

$$B(x_c, y_c, r, I) = \left\{ I_{ij} |\ \ |(x_c - i)^2 + (y_c - j)^2 - r^2| \leq \left(\frac{1}{2}\right)^2 \right\}, \tag{6.3}$$

and the set of interior cell pixels

$$C(x_c, y_c, r, I) = \left\{ I_{ij} |\ (x_c - i)^2 + (y_c - j)^2 \leq \left(\frac{r}{2}\right)^2 \right\}, \tag{6.4}$$

Computing Sample Means and Probabilities of B and C Sets

We extract sample mean $\bar{B}$ from the set of boundary cell pixels

$$\bar{B} = \frac{\sum_i B_i}{|B|} \tag{6.5}$$

and sample means $\bar{C}$ from the set of interior cell pixels

$$\bar{C} = \frac{\sum_i C_i}{|C|} \tag{6.6}$$

where B_i and C_i are the i^{th} element of the respective set. Based on a visual examination of the distribution of sample points of $\bar{B}$ derived from real imagery, the probability density of cell boundary P_{cb} is modelled as Gaussian with mean μ_{cb} and variance σ_{cb}^2

$$P_{cb}(\bar{B}) \sim N(\bar{B}; \mu_{cb}, \sigma_{cb}^2), \tag{6.7}$$

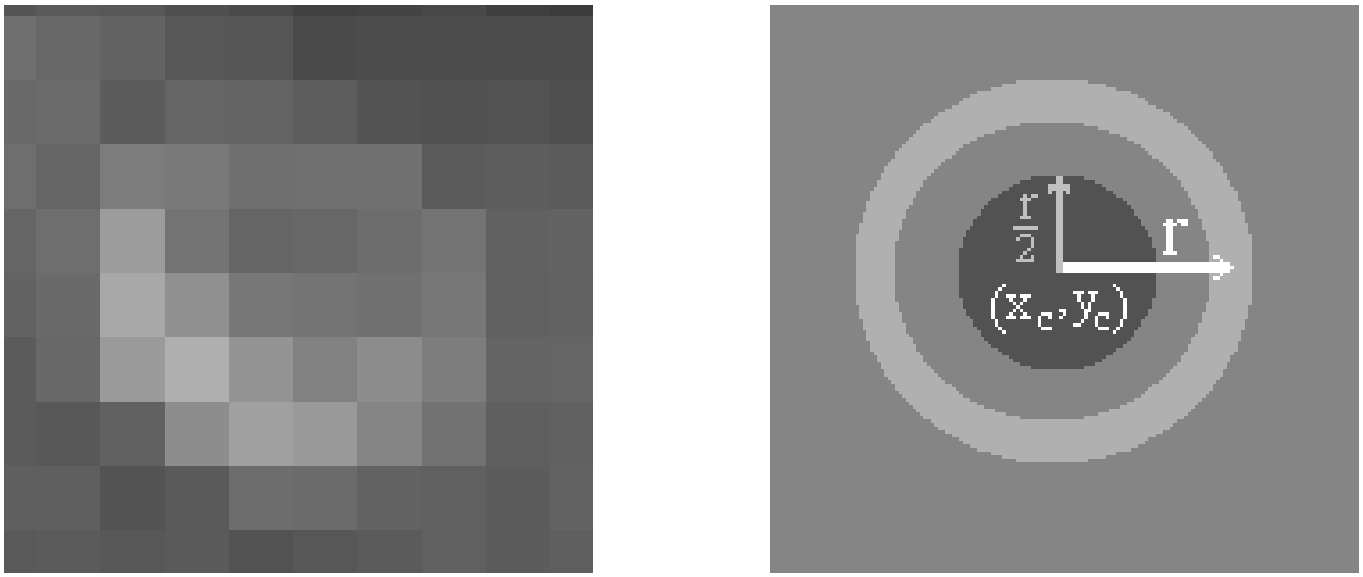


Figure 6.4: (a) 8 by 8 pixel detail of a HSC phase contrast microscope image. (b) A circular idealized cell model.

where μ_{cb} and σ^2_{cb} are estimated empirically. Similarly the probability density of dark region inside the cell P_{ci} is also modelled as Gaussian with mean μ_{ci} and variance σ^2_{ci}

$$P_{ci}(\bar{C}) \sim N(\bar{C}; \mu_{ci}, \sigma^2_{ci}), \tag{6.8}$$

where μ_{ci} and σ^2_{ci} are again estimated empirically.

It should be mentioned that the parameters of model (6.2) are time invariant, consistent with most of our acquired data sets. Therefore in cases where the intensity or contrast of the image changes over time due to background noise or spatio-temporal illumination variations, the non-stationarity of the data might make (6.2) in error or inapplicable. In such cases, to improve the robustness of the proposed method, the time variations of the image sequence need to be removed by background estimation and subtraction, considered in the next chapter.

Penalizing Spurious Cell Detection

As illustrated in Fig. 6.5 we wish to penalize spurious cell detection. We propose to calculate an empirical cumulative density function (CDF) to discriminate background from cell boundary. The CDF on cell boundary pixel intensities is computed by

$$cdf_n(B) = \frac{\sum_{i=1}^{n} B_i}{|B| \cdot \bar{B}},\ n \in 1 : |B| \tag{6.9}$$

As a set of constant or uniform values in B corresponds to a straight line CDF, we use a Kolmogorov-Smirnov [119] test on B to test its deviation from uniformity:

$$D(cdf) = \max_{n \in [1:N]} |cdf_n - \frac{n}{N}| \tag{6.10}$$

An exponential function $P_{bu}(D)$ is used to penalize the non uniformity as

$$P_{bu}(D) = \exp\{-2 \cdot N \cdot D(cdf)\} \tag{6.11}$$

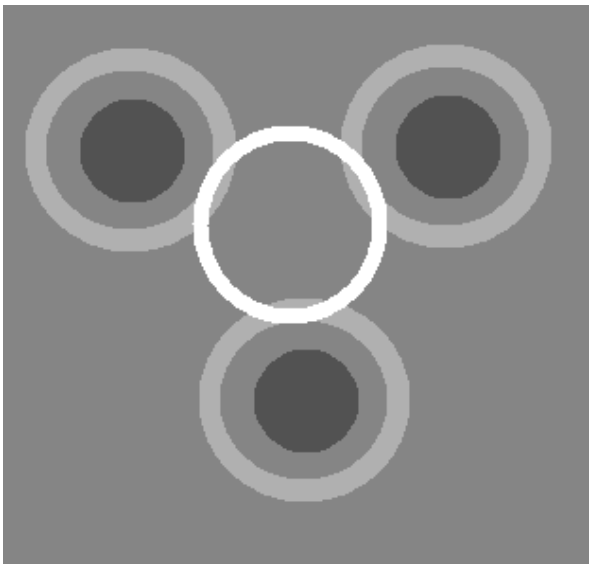


Figure 6.5: A scenario in which a spuriously hypothesized (white) cell may have a large associated average brightness $\bar{B}$ and a low cell interior brightness $\bar{C}$. The uniformity constraint (HSC33) in the cell boundary is intended to address this case.

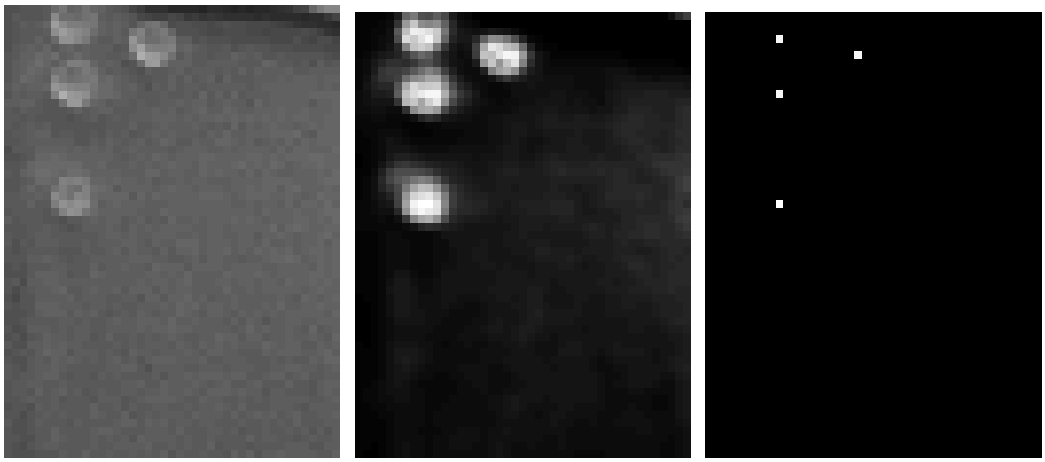
Locating The Cell Centers

The development of a simple model for the detection of cells in background noise is completed by combining cell criteria in (6.2). Finally, to locate the cell centers, cell model (6.2) is first applied to the given HSC image. We then find the local maxima in P (6.2) and then threshold the local maxima map.

6.2.2 Results

Some results obtained by applying the proposed cell model in (6.2) to the original HSC images of Phenotype 1 follow. As can be observed in the figures 6.6 to 6.9, applying the initial probabilistic cell model to HSC images of Phenotype 1, it is able to identify both non-dividing and dividing cell centers correctly.

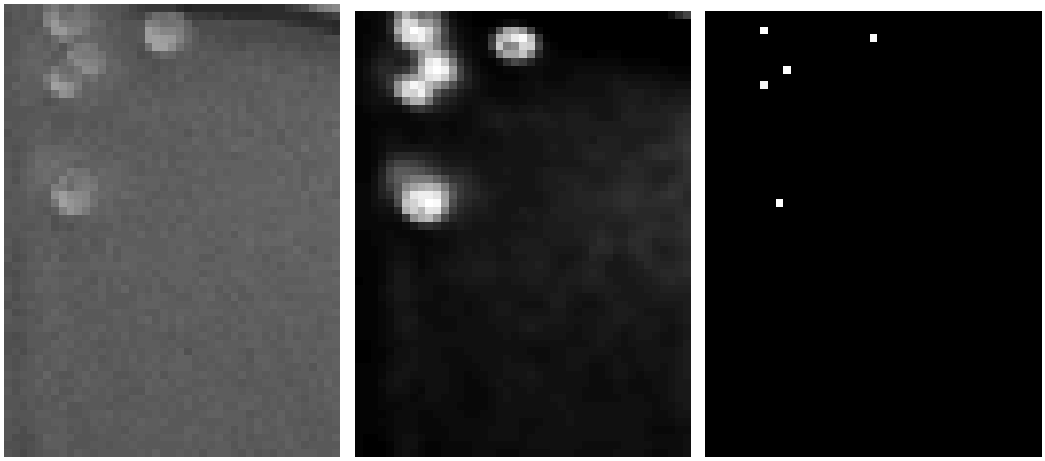
HSC Phenotype 1 - Non-dividing cells



(a) (b) (c)

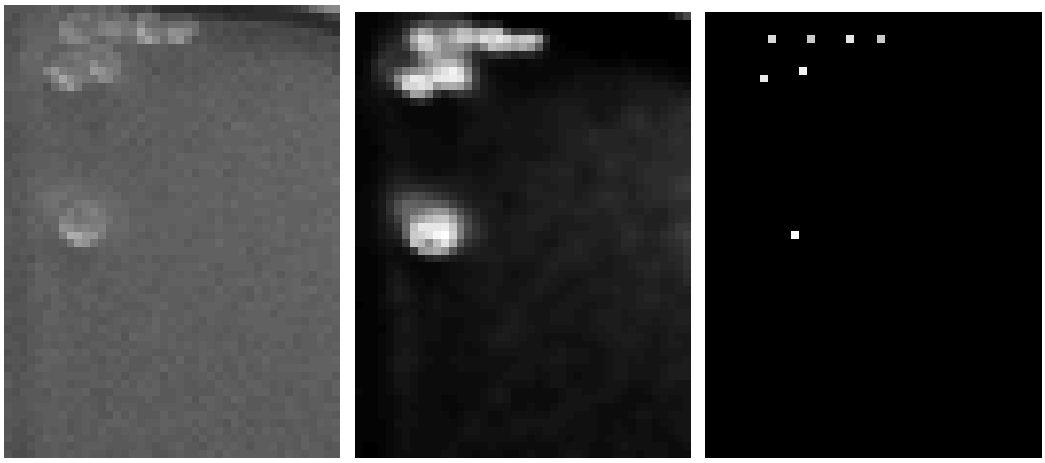
Figure 6.6: (a) Microscope image I. (b) Probability map $P(z|\ I, r_\circ)$ obtained by applying cell model (6.2). (c) Thresholding the local maxima map of $P(z|\ I, r_\circ)$.

HSC Phenotype 1 - Dividing cells



(a) (b) (c)

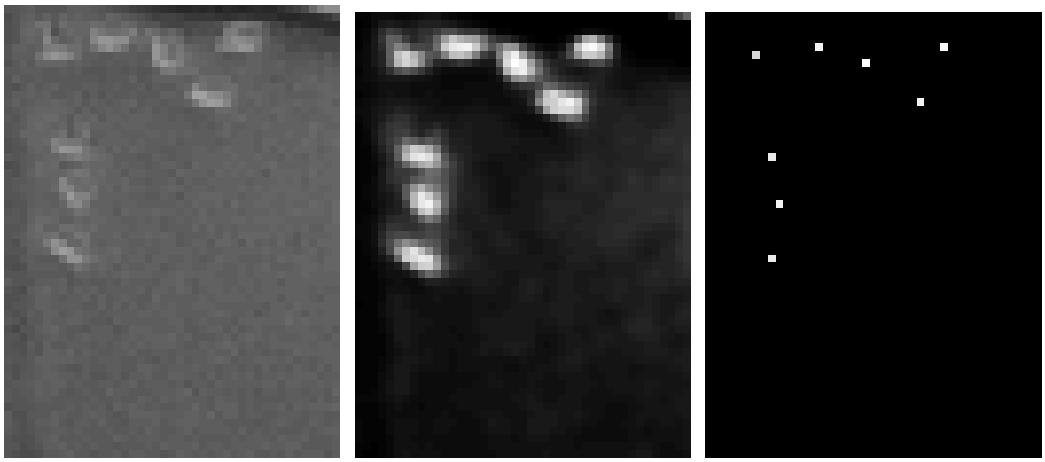
Figure 6.7: (a) Microscope image I. (b) Probability map $P(z|\ I, r_\circ)$ obtained by applying cell model (6.2). (c) Thresholding the local maxima map of $P(z|\ I, r_\circ)$.



(a) (b) (c)

Figure 6.8: (a) Microscope image I. (b) Probability map $P(z|\ I, r_\circ)$ obtained by applying cell model (6.2). (c) Thresholding the local maxima map of $P(z|\ I, r_\circ)$.

HSC Phenotype 1 - Divided cells



(a) (b) (c)

Figure 6.9: (a) Microscope image I. (b) Probability map $P(z|\ I, r_\circ)$ obtained by applying cell model (6.2). (c) Thresholding the local maxima map of $P(z|\ I, r_\circ)$.

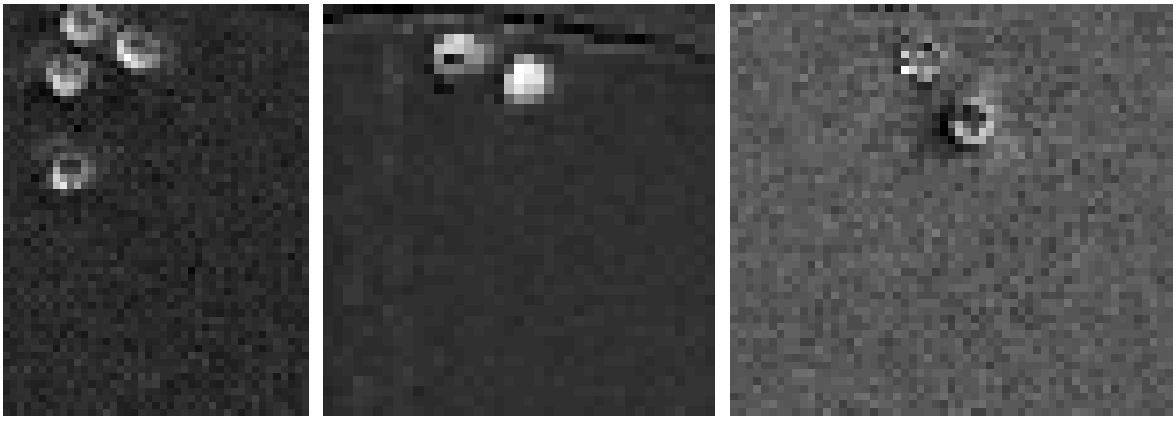
6.3 Thresholding Cell Detection

Three different HSC phenotypes are depicted in Fig. 6.10. To be used for detection of all HSC phenotypes investigated in this work, the proposed method in this section is a more generalized approach in comparison with the method that was proposed in Sec. 6.2. The description follows.

6.3.1 Feature Set

As we can see in Fig. 6.10, except for Phenotype 1, the other HSC phenotypes cannot be modelled as an object with dark interior and bright boundary (6.2), therefore the proposed method in Sec. 6.2 performs poorly to detect HSCs of Phenotype 2 and Phenotype 3.

To design a general method that could be applicable for detecting different HSC pheno-



Phenotype 1 Phenotype 2 Phenotype 3

Figure 6.10: Different HSC phenotypes: (Phenotype 1) Cells have bright boundary and dark interior. (Phenotype 2) Cells are completely bright. (Phenotype 3) Poor contrast cells.

types investigated in this research, some common features among different HSC phenotypes must be extracted. All HSC phenotypes in this work can be characterized as an approximately circular object. Cell pixels have also high intensity variations against a uniform background, hence a HSC can be localized by detecting a group of pixels located in a circle and have high intensity variations against a uniform background. These criteria can be summarized as follows.

1. The cell is round, with some radius,

2. Cell pixels are significantly different from the uniform background pixels.

So to locate HSCs, for each pixel (x, y) in the image, the sample mean of square intensities of all its neighbour pixels located in a circle with center coordinates (x, y) and radius r is computed. A binary image is then generated by segmenting the mean square image to foreground and background. Having a binary image, Euclidean distance of foreground

objects from the background is computed that will be referred as Euclidean image. As the last step, to localize the cell centers, the local maxima in the Euclidean image are found and thresholded.

Circular Mean Square

HSCs are modelled as a circular anomaly which is represented by a set of pixels with significant intensity variations against the uniform background. Assuming (x, y) and r as center coordinates and radius of a cell respectively, we construct the set $G(z_k^m, I_k)$, which returns the inside cell pixels

$$G(z, I) = \{I_{ij}|\ (x-i)^2 + (y-j)^2 \leq (r)^2\ \} \tag{6.12}$$

from which we extract the sample mean of square intensities

$$\bar{G} = \frac{\sum_{g \in G} g^2}{|G|} \tag{6.13}$$

Two Class Classification: Cell and Background

To recognize cells from the uniform background, first (6.13) is applied to the cell image and $\bar{G}$ is computed. The variance of the pixels $\{g|\ g \in G\}$ located inside a ring with radius r is

$$\sigma^2 = \frac{\sum_{g \in G}(g-\mu)^2}{|G|} = \frac{\sum_{g \in G} g^2 - \sum_{[1,|G|]} \mu^2}{|G|} \tag{6.14}$$

after simplification we have

$$\bar{G} = \frac{\sum_{g \in G} g^2}{|G|} = \sigma^2 + \mu^2 \tag{6.15}$$

Thus for G located in the uniform background we find

$$\bar{G}_{bkg} = \frac{\sum_{g \in G} g^2}{|G|} = \sigma^2_{bkg} \tag{6.16}$$

whereas for G located inside a cell we have

$$\bar{G}_{cell} = \frac{\sum_{g \in G} g^2}{|G|} = \sigma^2_{cell} + \mu^2_{cell} \tag{6.17}$$

For all of the different cell phenotypes one or both of the σ^2_{cell}, μ^2_{cell} are significantly higher than those of the background, therefore $\bar{G}_{bkg} << \bar{G}_{cell}$ and as a result $\bar{G}$ can be used to detect HSCs in the uniform background by classifying to two classes, cell and background, by minimizing the inter-class variance

$$\sigma^2(T) = l_{cell}(T) \cdot \sigma^2_{cell}(T) + l_{bkg}(T) \cdot \sigma^2_{bkg}(T) \tag{6.18}$$

where $l_{bkg}(T)$ and $l_{cell}(T)$ are the number of pixels in the background and the cell classes, $\sigma^2_{bkg}(T)$, $\sigma^2_{cell}(T)$, and $\sigma^2(T)$ are variance of background, variance of cell class and inter-class variance considering the threshold (T). Replacing σ^2_{bkg} and σ^2_{cell} respectively from (6.16) and (6.17) in (6.18) we have

$$\sigma^2(T) = l_{cell}(T) \cdot (\bar{G}_{cell}(T) + \mu^2_{cell}(T)) + l_{bkg}(T) \cdot \bar{G}^2_{bkg}(T) \tag{6.19}$$

Distance of Anomalous Pixels from Background

Considering the HSC as a circular anomaly in the proposed method it can be concluded that the cell center has the maximum distance to the cell boundary in comparison with any other pixel in the cell area. Thus to fit a circular shape to the classified anomalous regions, we compute $D(cell_p, bkg_p)$, the Euclidean distance of each anomalous pixel $cell_p = (x_{cell}, y_{cell})$ from its closest background pixel $bkg_p = (x_{bkg}, y_{bkg})$

$$D(cell_p, bkg_p) = \sqrt{(x_{cell} - x_{bkg})^2 + (y_{cell} - y_{bkg})^2} \tag{6.20}$$

where

$$bkg_p = arg \left\{ \min_{bkg_p} D(cell_p, bkg_p) \right\} \tag{6.21}$$

Locating The Cell Centers

We compute the product of the Euclidean distance map and the circular mean square

$$P_{cell} = D_{cell} \cdot \bar{G} \tag{6.22}$$

therefore from pixels with the same circular mean square value in $\bar{G}$, the one that is located closer to the centroid of a segmented cell region and so has a higher value in D_{cell}, will have higher value in P_{cell} and as a result will be more likely to be a cell center. Finally, to locate the cell centers, we find the local maxima in P_{cell} and then threshold the local maxima map.

6.3.2 Results

Some results obtained by applying the proposed cell detection method in (6.22) to HSC images of different phenotypes follow. As can be observed in the figures 6.11 to 6.19, applying the proposed thresholding cell detection method to HSC images of different phenotypes investigated in this work, it is able to identify both non-dividing and dividing cell centers correctly.

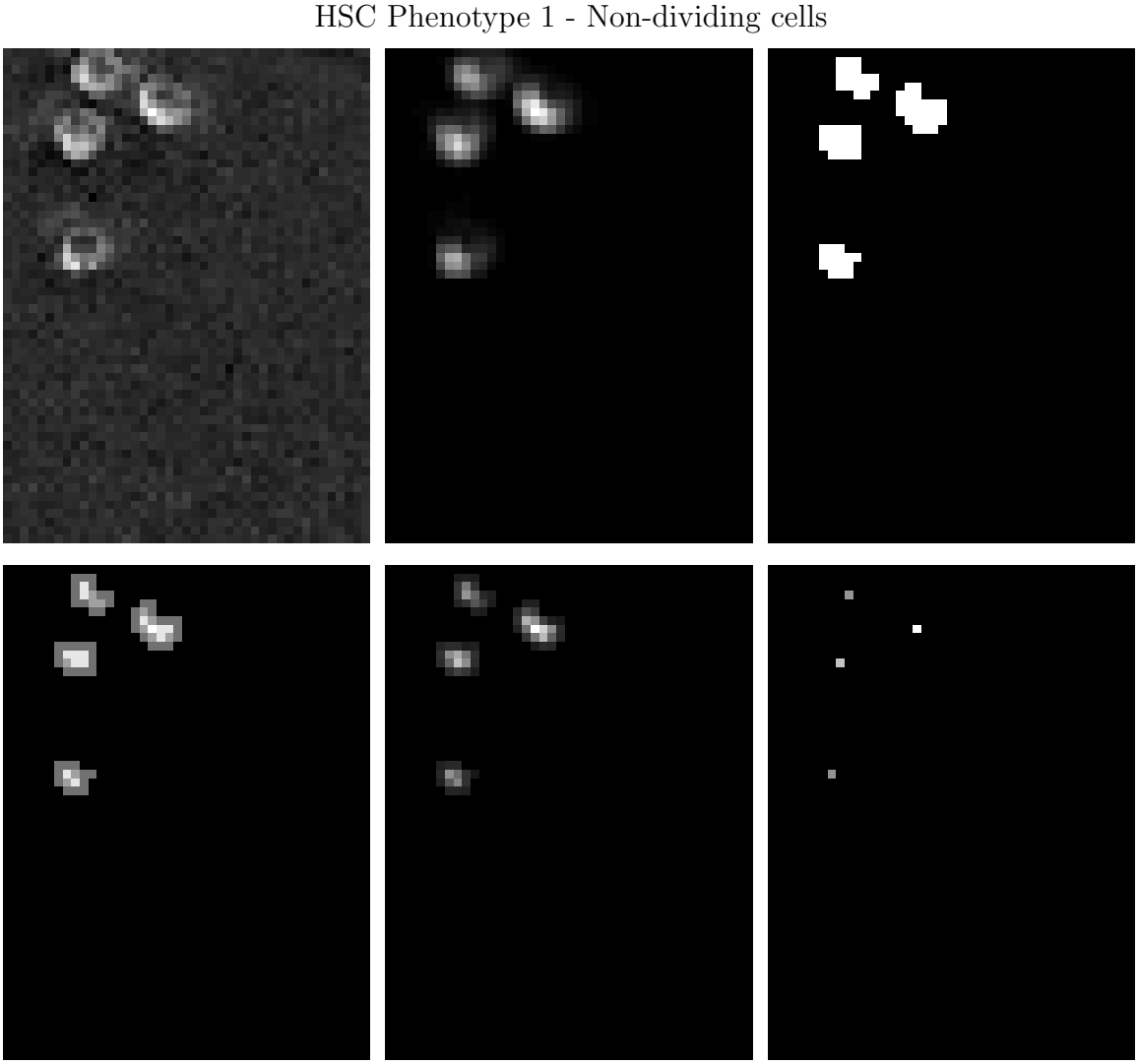


Figure 6.11: (a) Original HSC image. (b) Circular mean square. (c) Classification of circular mean square to cell and background classes by minimizing the inter-class variance. (d) Euclidean distance of cell pixels from the background. (e) Product of Circular mean square and Euclidean distance. (f) Cell center locations after thresholding the maxima map.

HSC Phenotype 1 - Dividing cells

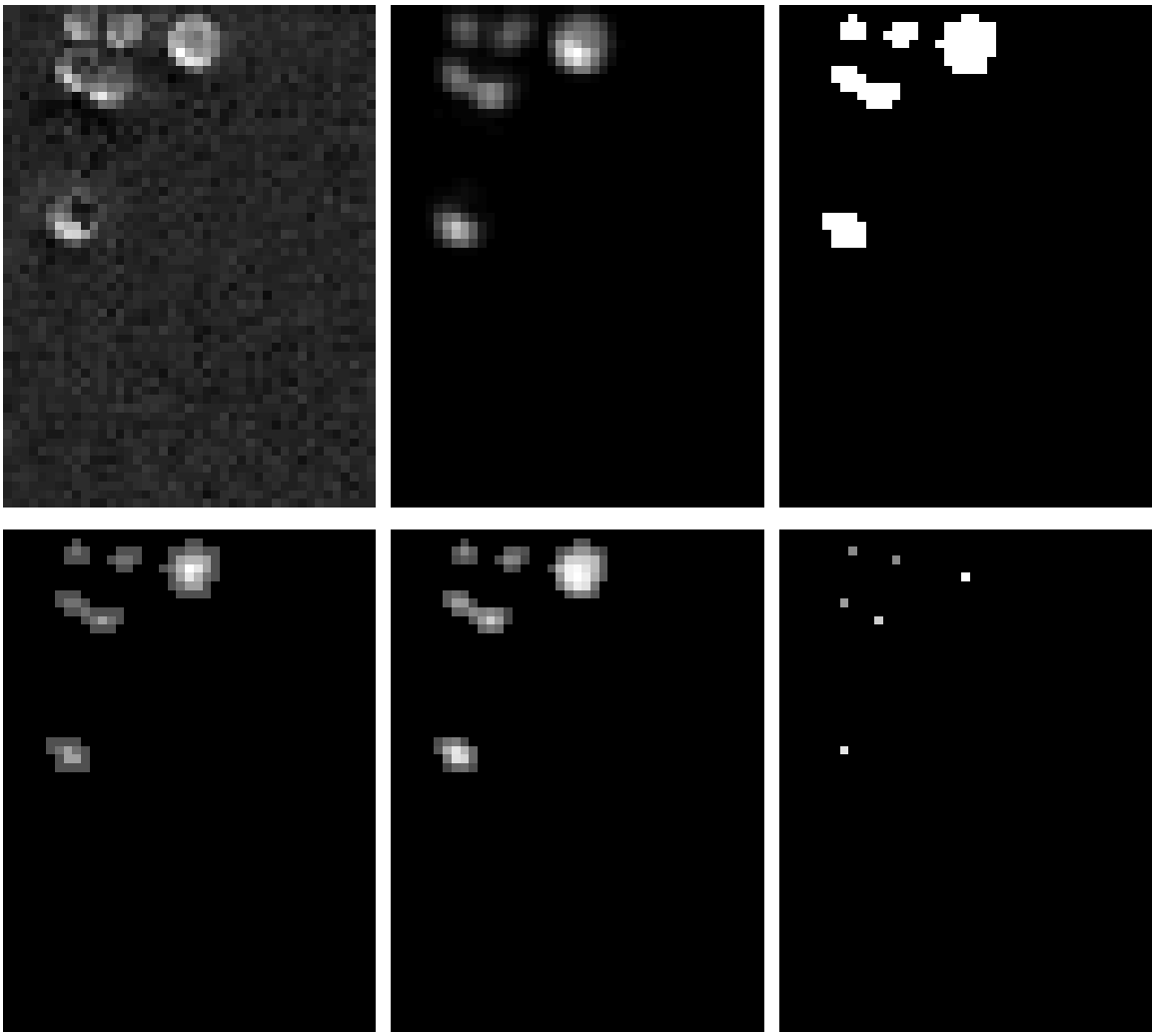


Figure 6.12: (a) Original HSC image. (b) Circular mean square. (c) Classification of circular mean square to cell and background classes by minimizing the inter-class variance. (d) Euclidean distance of cell pixels from the background. (e) Product of Circular mean square and Euclidean distance. (f) Cell center locations after thresholding the maxima map.

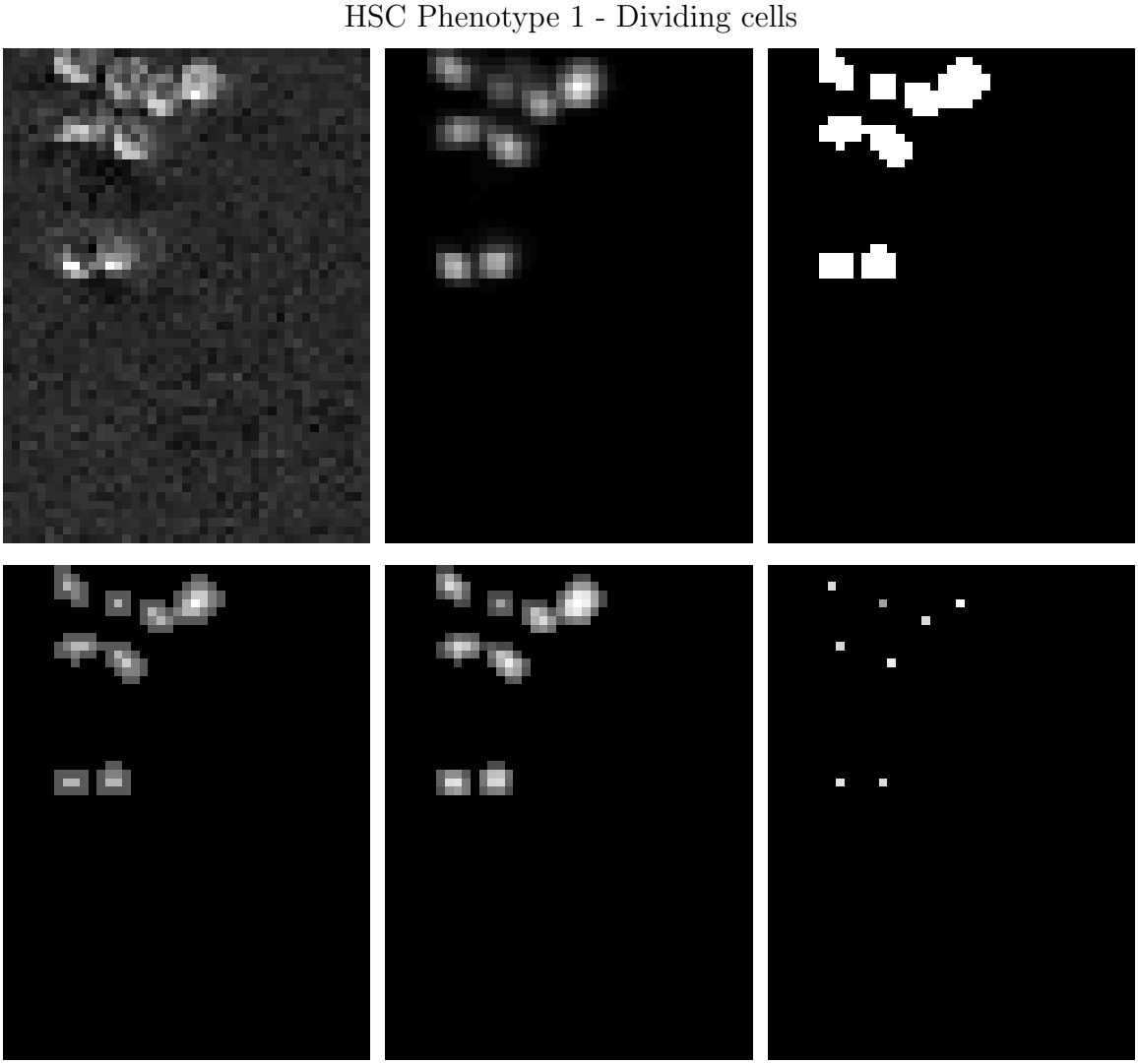


Figure 6.13: (a) Original HSC image. (b) Circular mean square. (c) Classification of circular mean square to cell and background classes by minimizing the inter-class variance. (d) Euclidean distance of cell pixels from the background. (e) Product of Circular mean square and Euclidean distance. (f) Cell center locations after thresholding the maxima map.

HSC Phenotype 2 - Non-dividing cells

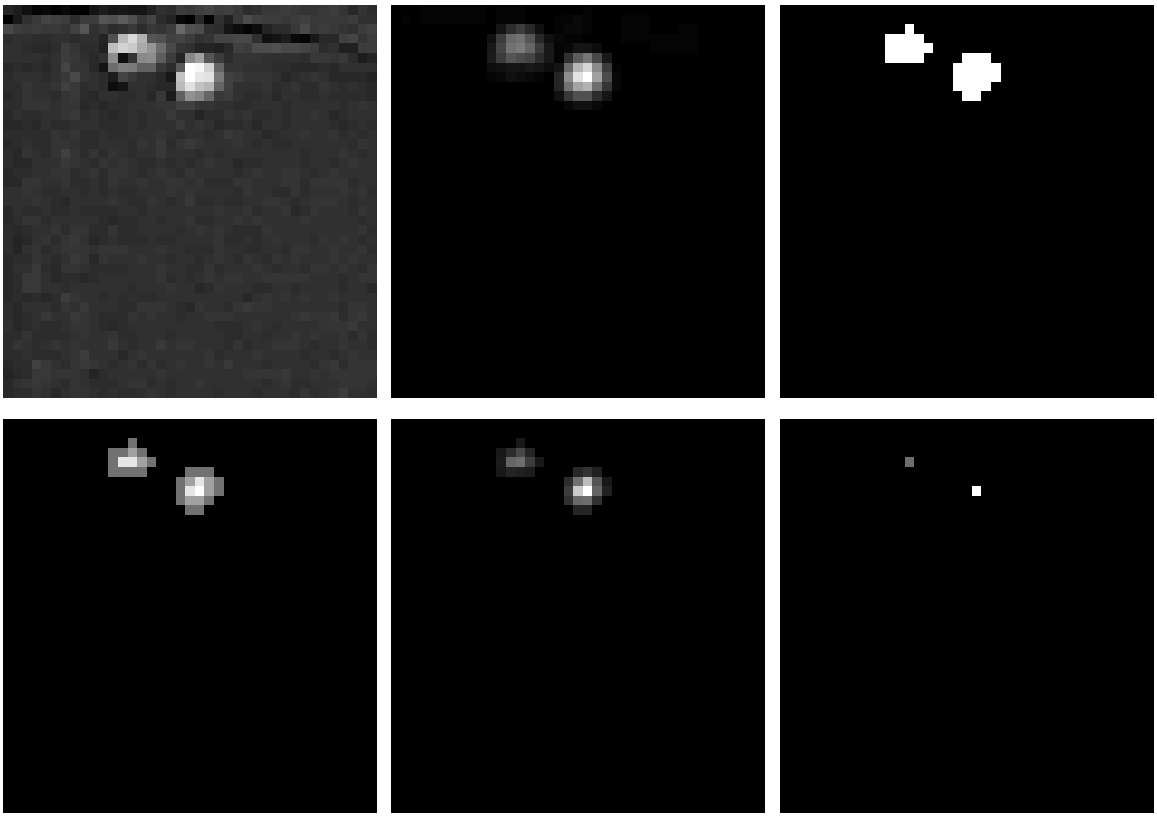


Figure 6.14: (a) Original HSC image. (b) Circular mean square. (c) Classification of circular mean square to cell and background classes by minimizing the inter-class variance. (d) Euclidean distance of cell pixels from the background. (e) Product of Circular mean square and Euclidean distance. (f) Cell center locations after thresholding the maxima map.

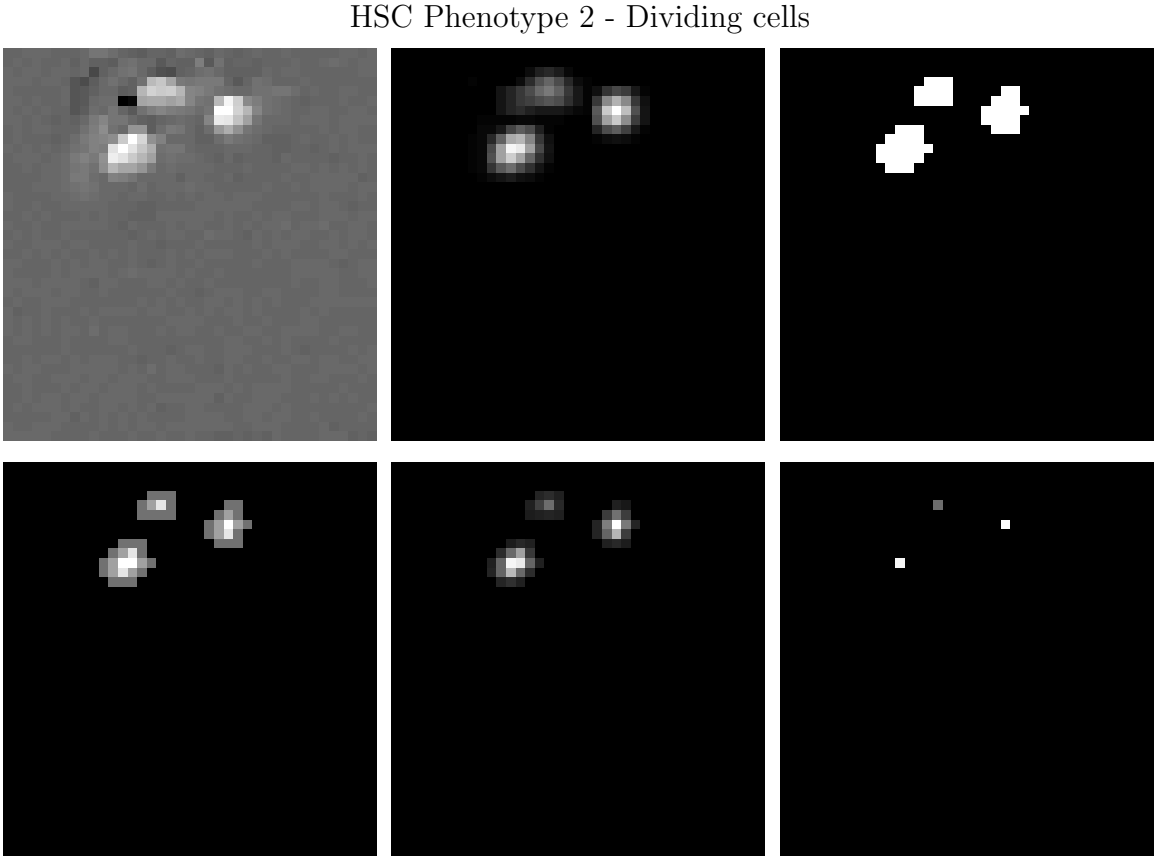


Figure 6.15: (a) Original HSC image. (b) Circular mean square. (c) Classification of circular mean square to cell and background classes by minimizing the inter-class variance. (d) Euclidean distance of cell pixels from the background. (e) Product of Circular mean square and Euclidean distance. (f) Cell center locations after thresholding the maxima map.

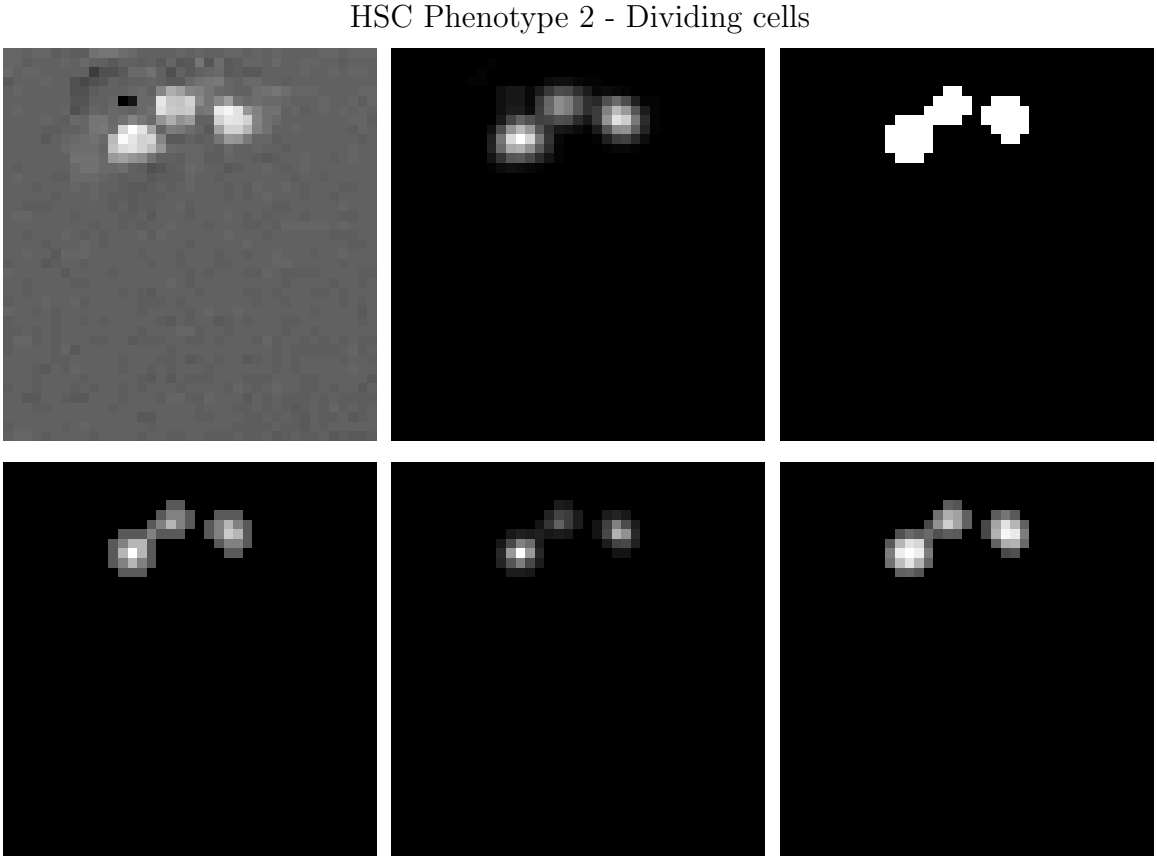


Figure 6.16: (a) Original HSC image. (b) Circular mean square. (c) Classification of circular mean square to cell and background classes by minimizing the inter-class variance. (d) Euclidean distance of cell pixels from the background. (e) Product of Circular mean square and Euclidean distance. (f) Cell center locations after thresholding the maxima map.

HSC Phenotype 3 - Non-dividing cells

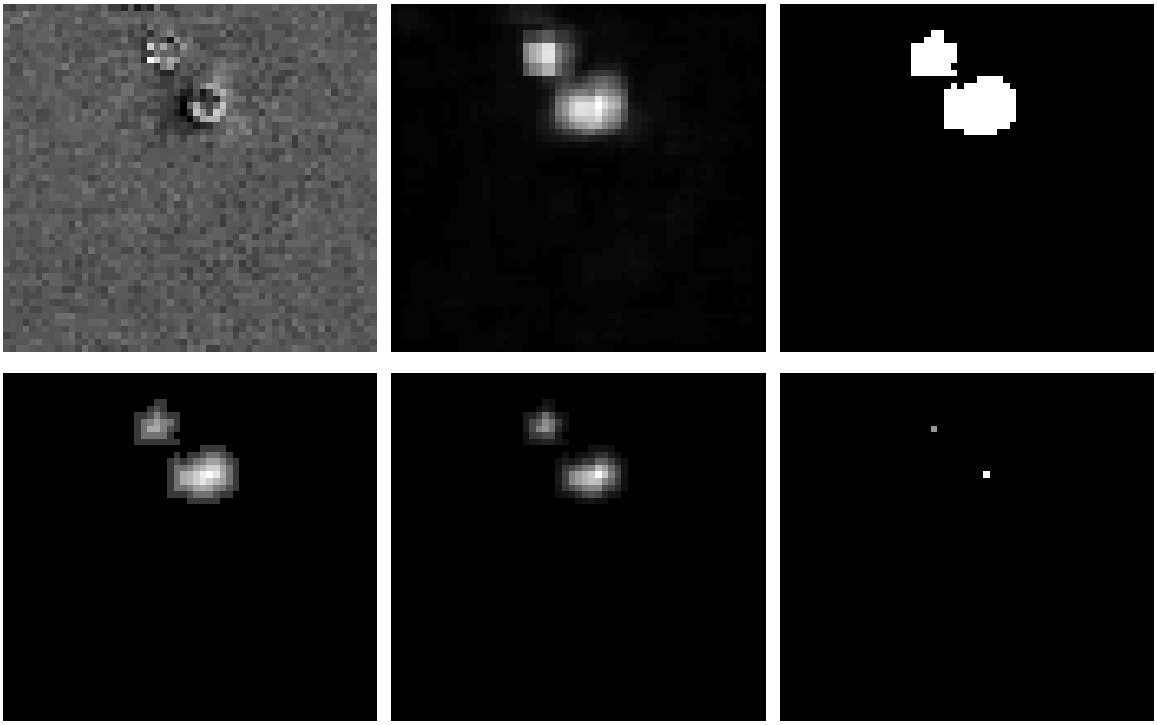


Figure 6.17: (a) Original HSC image. (b) Circular mean square. (c) Classification of circular mean square to cell and background classes by minimizing the inter-class variance. (d) Euclidean distance of cell pixels from the background. (e) Product of Circular mean square and Euclidean distance. (f) Cell center locations after thresholding the maxima map.

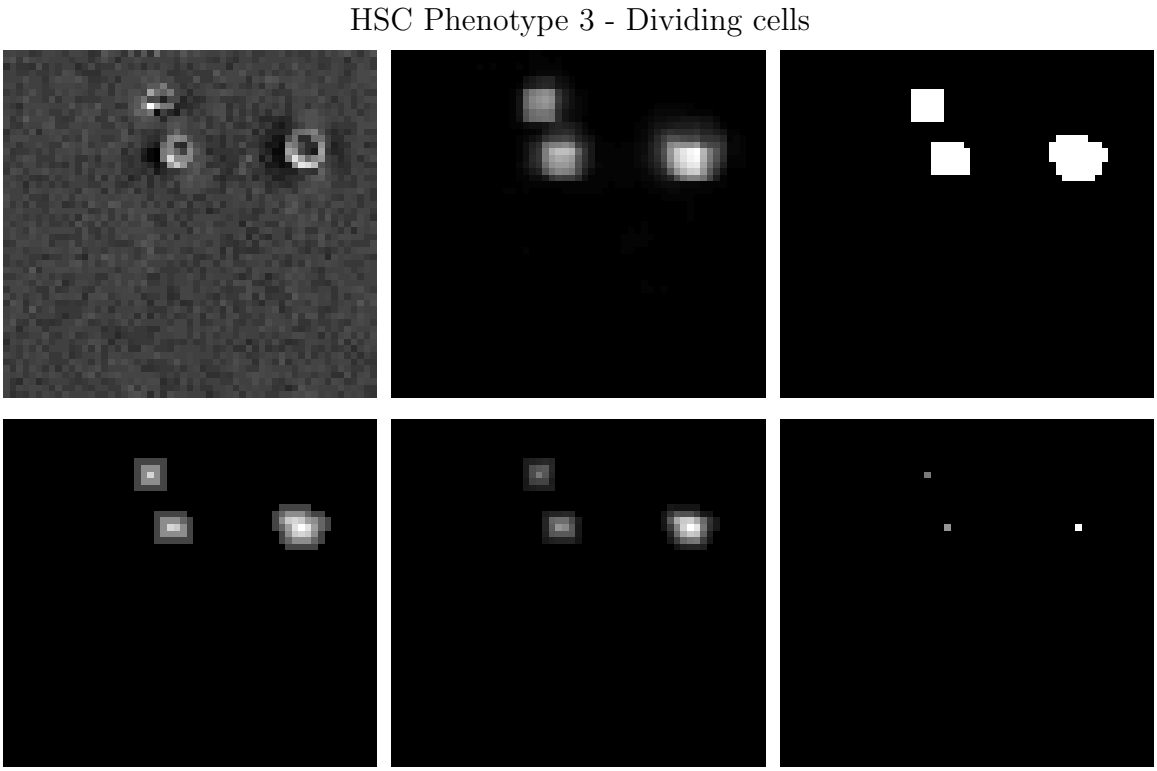


Figure 6.18: (a) Original HSC image. (b) Circular mean square. (c) Classification of circular mean square to cell and background classes by minimizing the inter-class variance. (d) Euclidean distance of cell pixels from the background. (e) Product of Circular mean square and Euclidean distance. (f) Cell center locations after thresholding the maxima map.

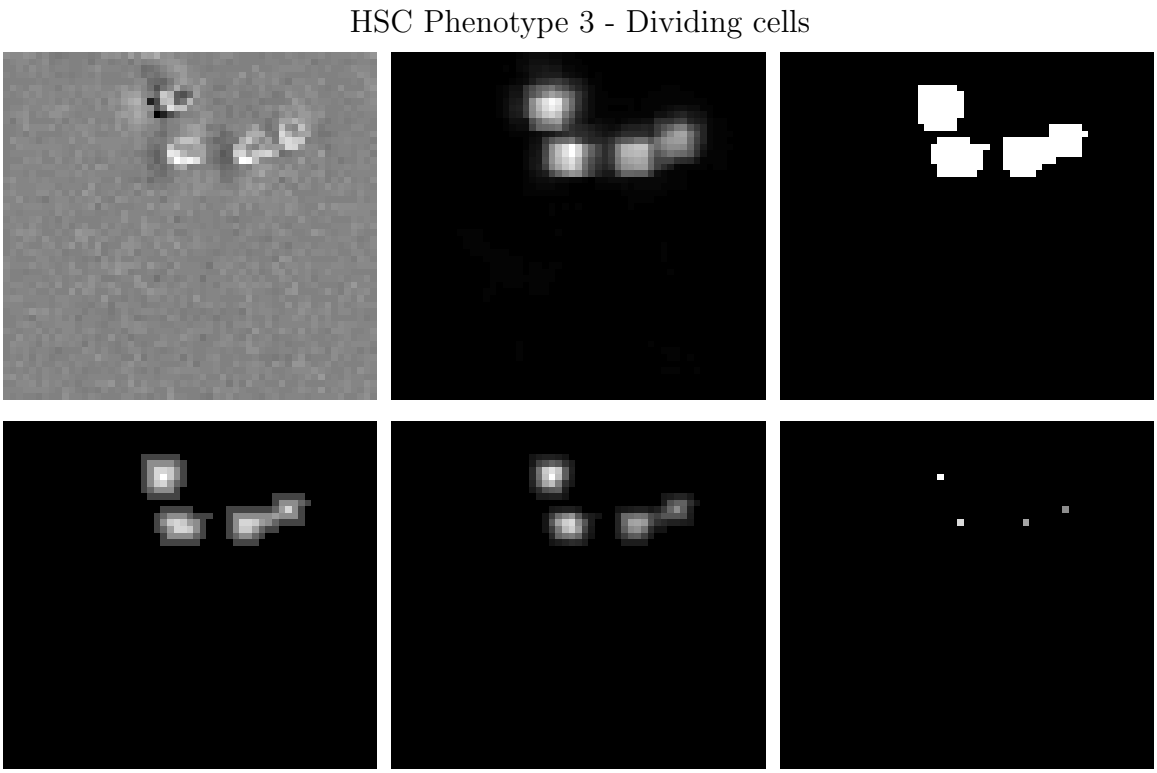


Figure 6.19: (a) Original HSC image. (b) Circular mean square. (c) Classification of circular mean square to cell and background classes by minimizing the inter-class variance. (d) Euclidean distance of cell pixels from the background. (e) Product of Circular mean square and Euclidean distance. (f) Cell center locations after thresholding the maxima map.

6.4 Probabilistic Cell Detection

The proposed approach in Sec. 6.3 is a general approach which is applicable to discriminate HSCs of different phenotypes investigated in this work from the uniform background. However, the proposed method in Sec. 6.3 may potentially perform better to discriminate single cells and dividing cells from each other, if some other features could be extracted and integrated in the model.

6.4.1 Feature Set

To design a probabilistic cell model similar to (6.2) in Sec. 6.2, but generalized to all HSC phenotypes which are investigated in this work similar to the proposed method in Sec. 6.3, we propose a cell model that integrates the following features.

As we mentioned in Sec. 6.3 and is depicted in Fig. 6.20, all of the HSC phenotypes can be characterized as approximately circular objects with high intensity variations against the uniform background. Moreover as we can observe in Fig. 6.20, most of the pixels around a cell are near the background mean, and the cell brightness pattern is symmetric about the cell center. These features are combined in a new cell model to discriminate cells from the uniform background and to better discriminate single cells from dividing ones. The proposed new probabilistic cell model is the product of three probabilistic terms: P_{cell}, P_{pnlz}, and P_{opp} which are associated with the cell probability, the penalizing probability, and the discrimination probability respectively.

$$P(z_k^m|\ I_k) = P_{cell}(z_k^m|\ I_k) \cdot P_{pnlz}(z_k^m|\ I_k) \cdot P_{opp}(z_k^m|\ I_k) \tag{6.23}$$

The probabilistic cell model is a function of $z = (x, y, r)$, where (x, y) is cell center location, and r is the cell radius. This revised model (6.23) is similar in sprit to (6.2), but generalized

to all HSC phenotypes investigated in this work. The description and computation of each probabilistic term follows.

Cell Probability as a Circular Anomaly

Recall (6.13)

$$\bar{G} = \frac{\sum_{g \in G} g^2}{|G|} \tag{6.24}$$

that was developed in Sec. 6.3 to extract the sample mean of square intensities of inside cell pixels. As it was discussed in Sec. 6.3, $\bar{G}$ can effectively be employed to discriminate cell pixels from a uniform background. We propose the cell probability P_{cell} to be an exponential

$$P_{cell}(z_k^m |\ I_k) = 1 - exp\{-\bar{G}(z_k^m, I_k)\} \tag{6.25}$$

so that P_{cell} has a strong response for cell pixels whereas it has a poor response (close to zero) for background pixels.

Penalizing False Candidates

As we can observe in Fig. 6.20 for all cell phenotypes, intensity variations is noticeable for all pixels which fall in the cells dividing region for proximate and splitting cells. This is true for all different HSC phenotypes depicted in Fig. 6.20 top, middle, and bottom. These pixels are located close to the cell boundary (the division line of two cells) and are not good candidate for the cell centers, however they will be considered as potential cell candidate as circular anomaly with specific radius against the uniform background based on their strong cell probabilities (P_{cell}). Therefore to discriminate a cell as circular anomaly from the background we construct a set of background pixels which are located outside the inner cell ring and inside an outer cell ring concentric with the cell, i.e., background pixels

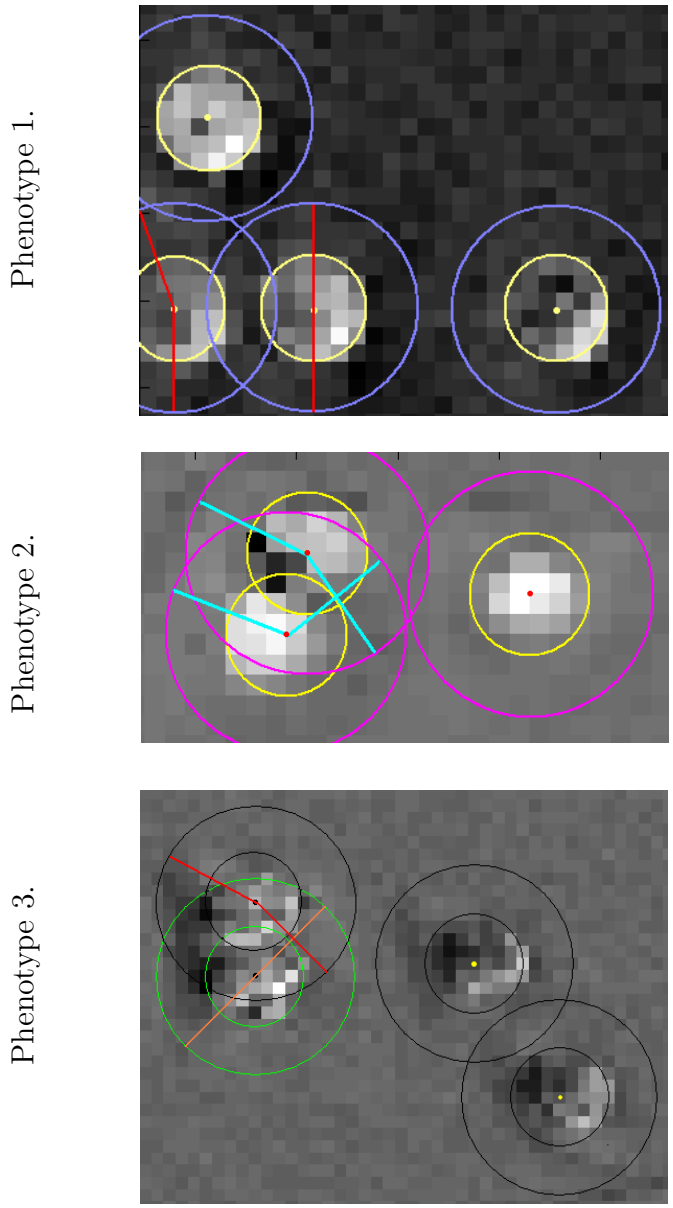


Figure 6.20: In the proposed cell model (6.23) in Sec. 6.4 assuming a uniform background, a cell is modelled by the product of P_{cell}, the cell probability based on cell interior, P_{pnlz}, the penalizing probability based on outer cell ring, and P_{opp}, the discrimination probability based on pixel pairs located on opposite sides of cell center.

which are located between two rings with radii r and $a \cdot r$

$$E(z, I) = \{I_{ij} |\ r^2 \leq (x-i)^2 + (y-j)^2 \leq (a \cdot r)^2\ \} \tag{6.26}$$

where $1 < a < 2$.

Assume we have an image sequence with a uniform background, the proposed cell probability term might detect false candidates with strong (P_{cell}) for adjacent cells. These false candidates are usually located between cell inner and outer rings where the absolute pixel intensities are significantly different from the background intensity. So to penalize false cell candidates let E^{α} be a subset of E which is constructed containing pixels of set E with intensities closer to the uniform background

$$E^{\alpha} = \Big\{ I_{ij} \ni\ |I_{ij} - \mu_{bkg}| < \Delta_{\alpha} \Big\} \tag{6.27}$$

where Δ_{α} is the α percentile in sorted set Δ, and

$$\Delta = \Big\{ |I_k^{ij} - \mu_{bkg}| \Big\} \tag{6.28}$$

We have,

$$|E^{\alpha}| = \alpha \cdot |E| \tag{6.29}$$

and $0 < \alpha < 1$. We then calculate the mean square $\bar{E^{\alpha}}$

$$\bar{E^{\alpha}} = \frac{\sum_{e \in E^{\alpha}} e^2}{|E^{\alpha}|} \tag{6.30}$$

Assume we have a HSC image with a uniform background and the background is a Gaussian with zero mean. The false candidates with strong (P_{cell}) will be penalized by P_{pnlz} that is explained as

$$P_{pnlz}(z|\ I) \sim N(0, \bar{E^{\alpha}}) \tag{6.31}$$

We have found that for $a = 1.5$ and $\alpha = 0.5$ the proposed P_{pnlz} works effectively for the HSC data set being considered here, however a and α might be set to different values for other data sets.

Discrimination Probability of Opposite Pixel Pairs

The magnitude difference of pixel pairs located in the opposite sides of the cell center (inside the cell) is less than the magnitude difference of inside cell and background pixels. Therefore we construct the set of pixel pairs located in the opposite sides of the cell center as

$$O(z, I) = \left\{ (I_{x+i,y+j},\ I_{x-i,y-j}) |\ \ i^2 + j^2 \leq r^2 \right\} \tag{6.32}$$

from which we extract the mean square of intensity differences of pixel pairs located in the opposite sides of the cell center

$$\bar{O} = \frac{\sum_{(a,b)\in O}(a-b)^2}{|O|} \tag{6.33}$$

An exponential probability is proposed to discriminate inside cell pixels from outside cell pixels as

$$P_{opp}(z_k^m |\ I_k) = exp\{-\bar{O}(z_k^m, I_k)\} \tag{6.34}$$

so that P_{opp} has a strong response for cell pixels whereas it has a poor response for background pixels.

Locating The Cell Centers

The development of the proposed cell model for the detection of cells in a uniform background is completed by combining cell criteria in (6.23). To locate the cell centers, cell model (6.23) is first applied to the image frame. We then find the local maxima in P (6.23) and then threshold the local maxima map.

6.4.2 Results

Some results obtained by applying the proposed cell cell model in (6.23) to HSC images of different phenotypes follow. As can be observed in the figures 6.21 to 6.29, applying the proposed cell model to HSC images of different phenotypes investigated in this work, it is able to identify both non-dividing and dividing cell centers correctly.

HSC Phenotype 1 - Non-dividing cells

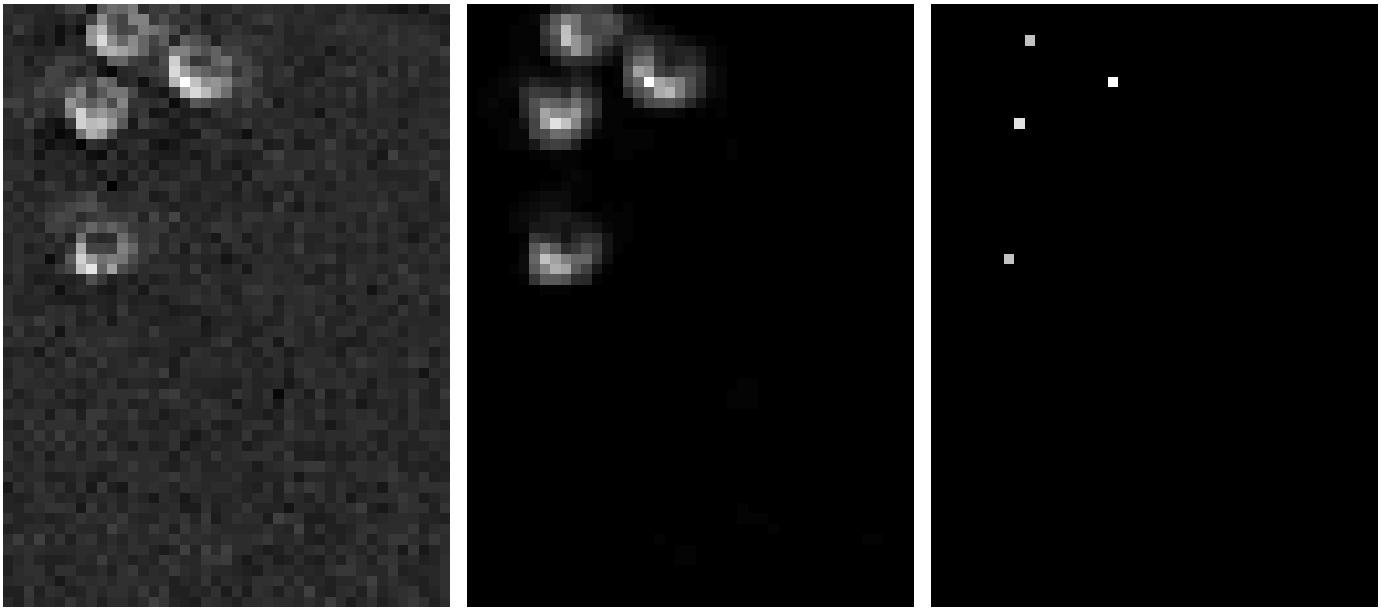


Figure 6.21: (a) Original HSC image. (b) Probability map. (c) Cell center locations after thresholding the maxima map.

HSC Phenotype 1 - Dividing cells

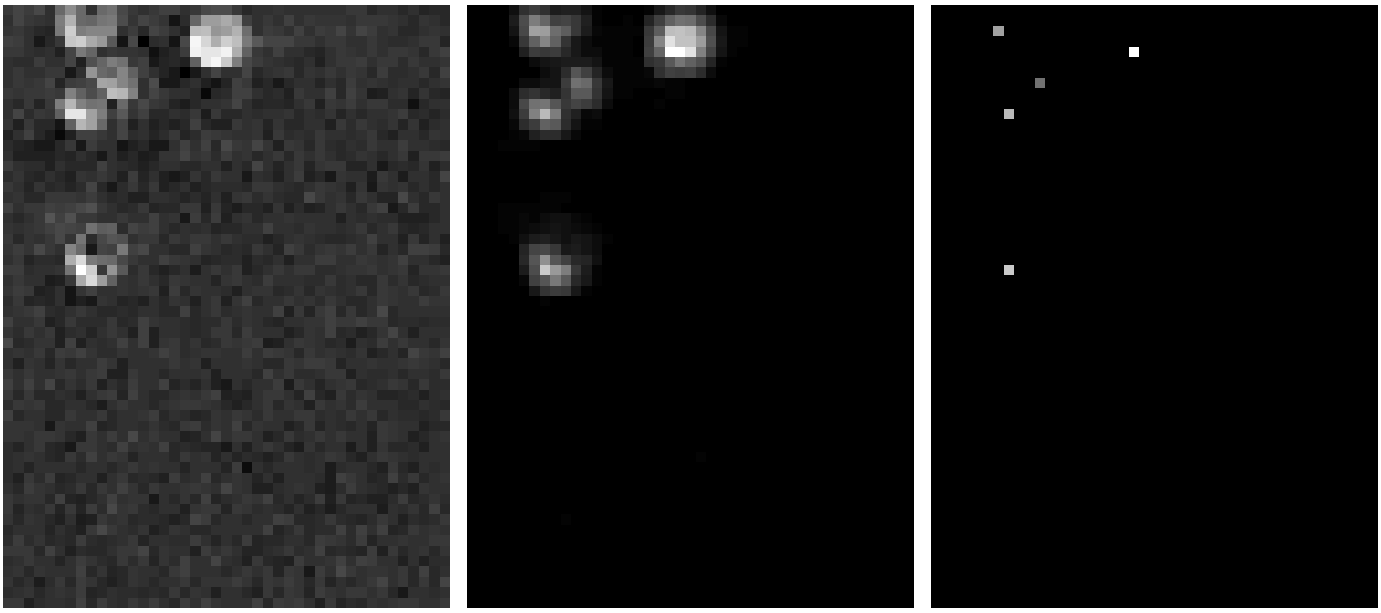


Figure 6.22: (a) Original HSC image. (b) Probability map. (c) Cell center locations after thresholding the maxima map.

HSC Phenotype 1 - Dividing cells

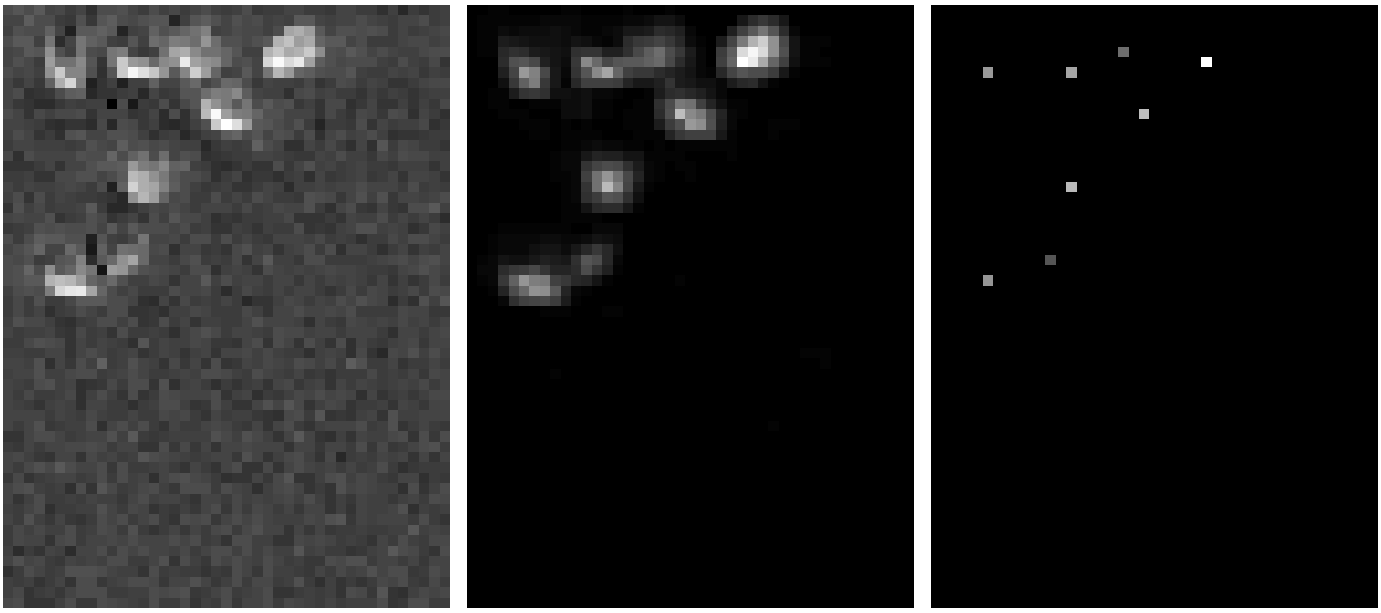


Figure 6.23: (a) Original HSC image. (b) Probability map. (c) Cell center locations after thresholding the maxima map.

HSC Phenotype 2 - Non-dividing cells

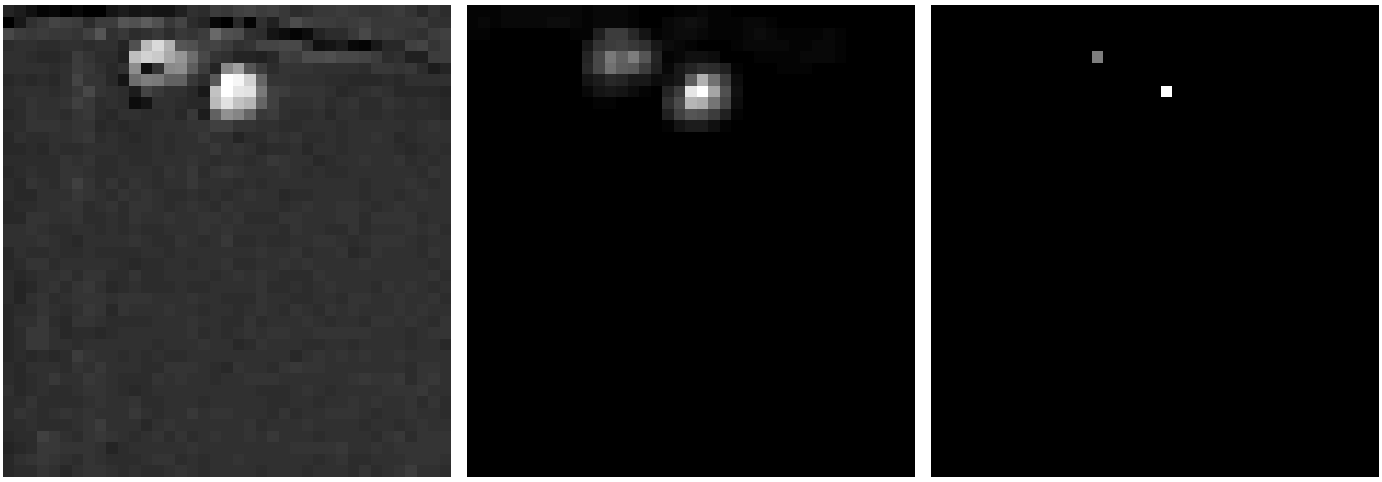


Figure 6.24: (a) Original HSC image. (b) Probability map. (c) Cell center locations after thresholding the maxima map.

HSC Phenotype 2 - Dividing cells

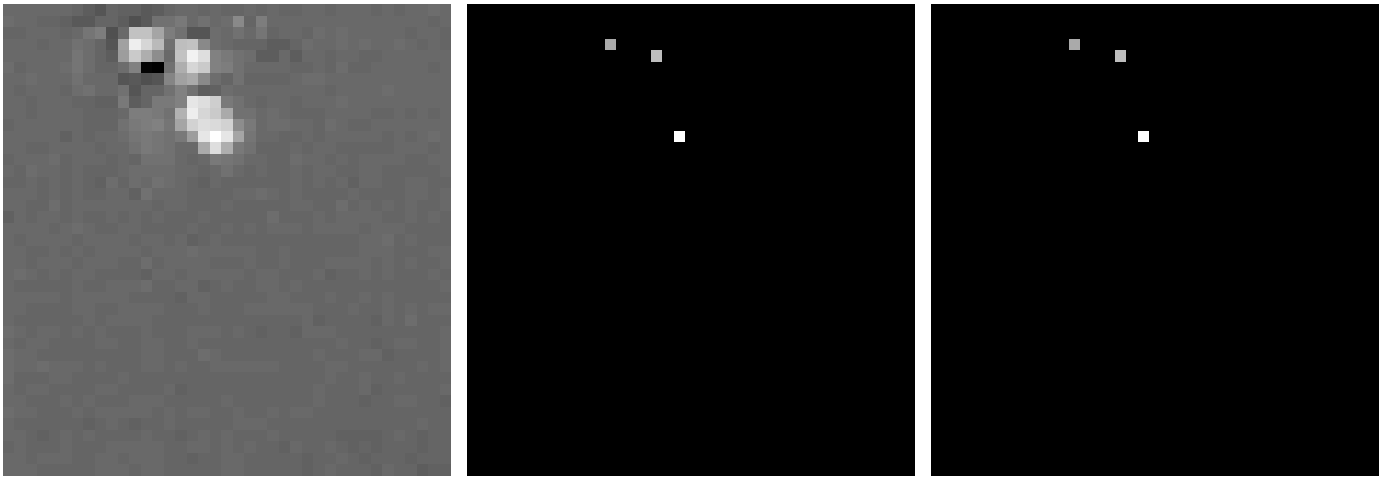


Figure 6.25: (a) Original HSC image. (b) Probability map. (c) Cell center locations after thresholding the maxima map.

HSC Phenotype 2 - Dividing cells

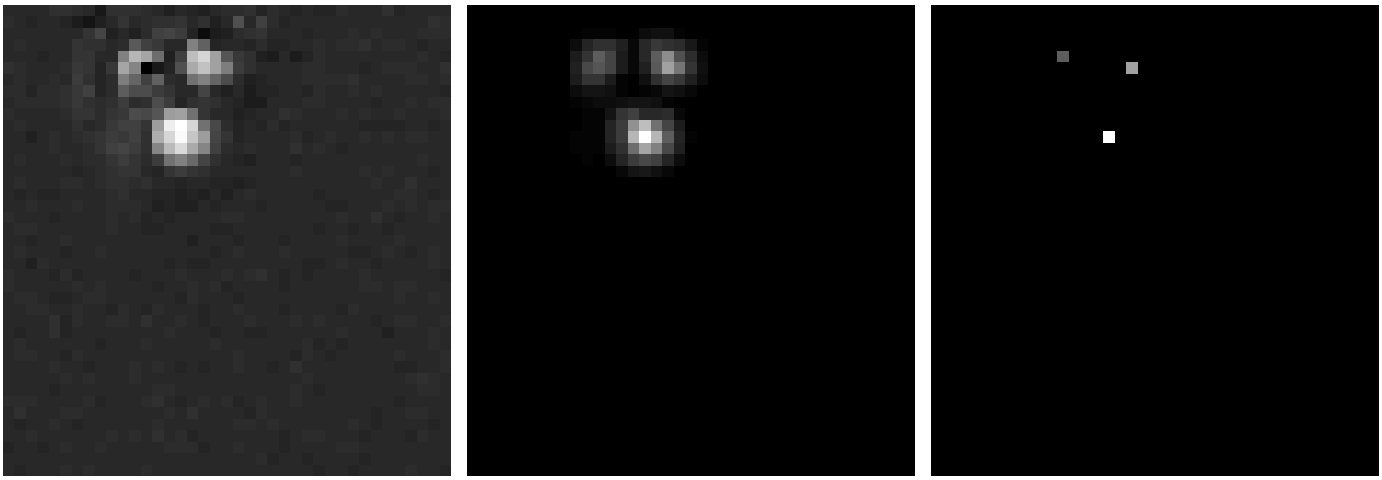


Figure 6.26: (a) Original HSC image. (b) Probability map. (c) Cell center locations after thresholding the maxima map.

HSC Phenotype 3 - Non-dividing cells

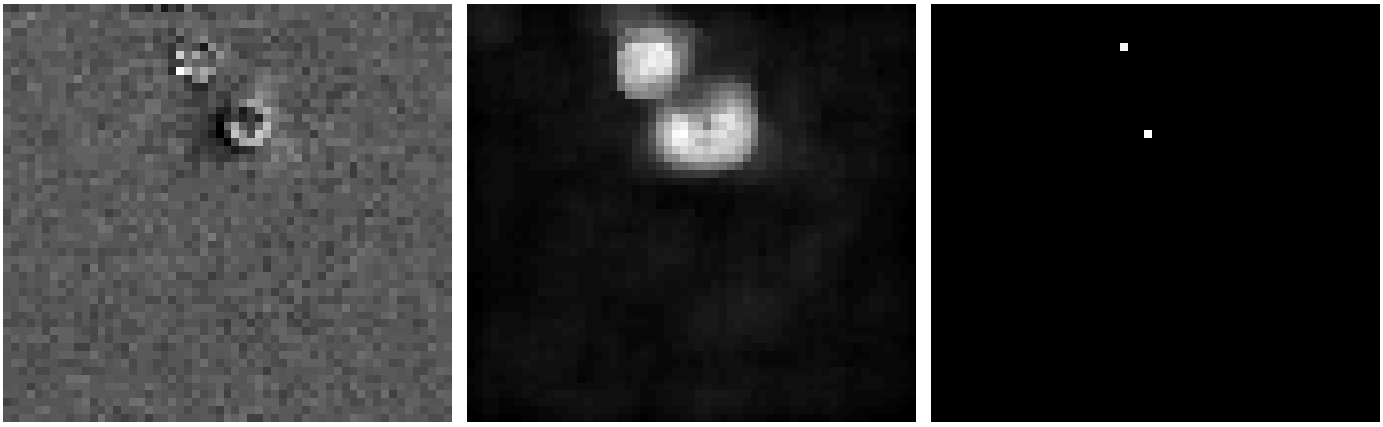


Figure 6.27: (a) Original HSC image. (b) Probability map. (c) Cell center locations after thresholding the maxima map.

HSC Phenotype 3 - Dividing cells

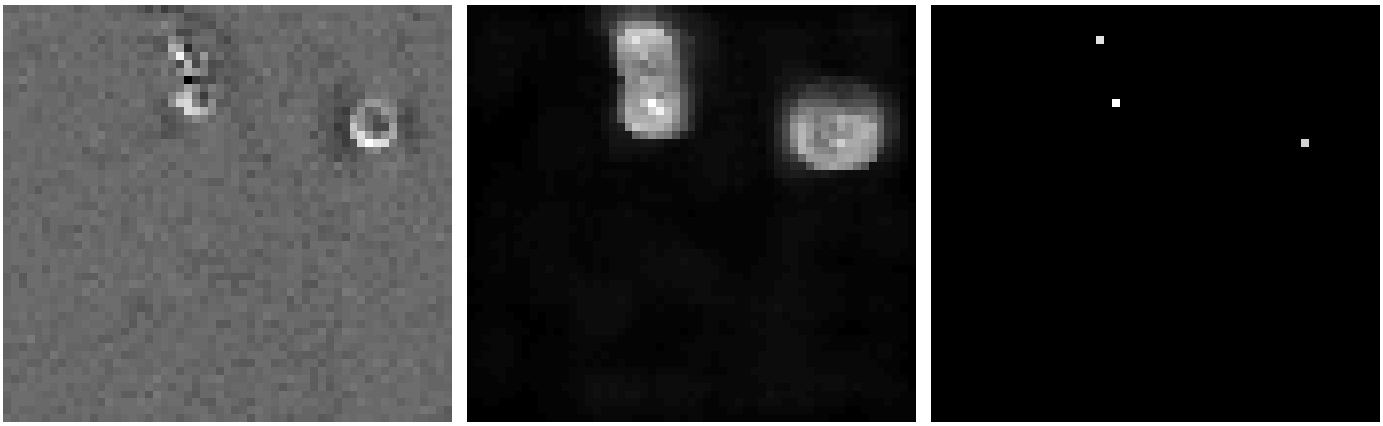


Figure 6.28: (a) Original HSC image. (b) Probability map. (c) Cell center locations after thresholding the maxima map.

HSC Phenotype 3 - Dividing cells

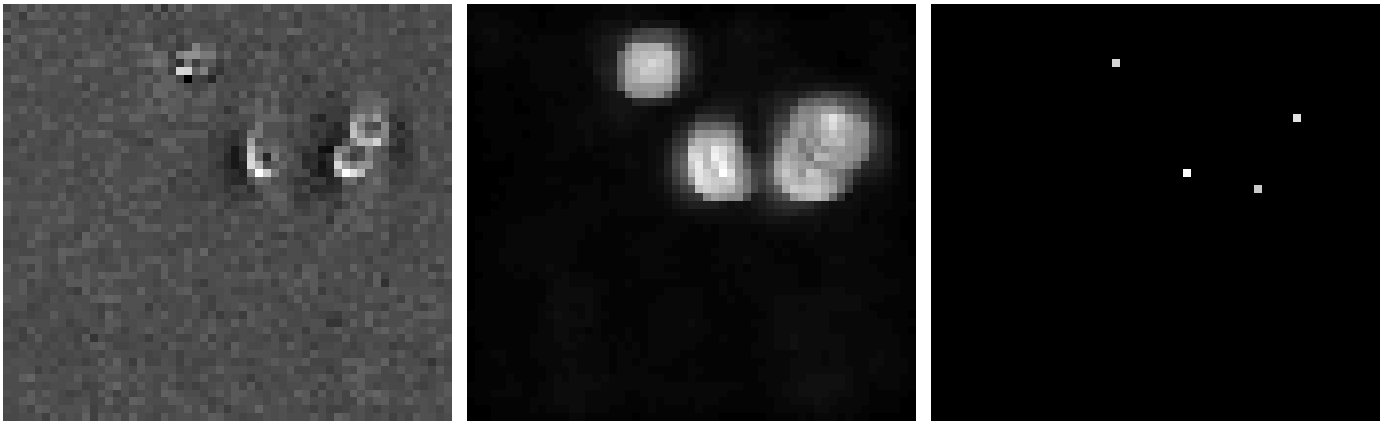


Figure 6.29: (a) Original HSC image. (b) Probability map. (c) Cell center locations after thresholding the maxima map.

6.5 A Deconvolution Model for Stem-Cell Localization

The methods discussed in Sections 6.2 and 6.3 perform well to locate non-dividing and dividing cells. As it was discussed in Sec. 6.4 a cell model may potentially perform better to discriminate single cells and dividing cells from each other, if some other features could be extracted and integrated in the model. Our concern in this section is to more specifically model dividing and close by cells.

6.5.1 The Proposed Method

Assume that we have segmented the cell areas containing individual or groups of cells and we want to locate the cell centre or cell centroid. Thus we essentially have an inverse problem which can be addressed in the form of a deconvolution problem such that a set of cell shape parameters must be found for optimal representation of cell segmented area. The proposed method solves the inverse problem using an optimized ellipse fitting method to find the optimal cell parameter set and locate the cell centres. This is a generic method, capable of modelling different cell types with changes in the model parameters, and robust against illumination variations. Our proposed method consists of cell template generation, template matching, and optimized ellipse fitting.

Elliptical Mean Square Model

A HSC can be discriminated as a group of pixels with significant intensity variations against a uniform background. HSCs have a fairly circular shape, however to design a general model an elliptical shape is employed to be used for segmentation of the cell region as a group of anomalous pixels. Let (c_x, c_y) be center coordinates, and a and b be horizontal

and vertical radii of the ellipse. The continuous elliptical cell is spatially discretized as

$$\frac{(x_l - c_x)^2}{a} + \frac{(y_l - c_y)^2}{b} \leq 1, \tag{6.35}$$

where (x_l, y_l) are coordinates of cell pixels. So the set of inside cell pixels can be explained by

$$C_p(c_x, c_y, a, b, I) = \{I_{ij} |\ \frac{(c_x - i)^2}{a} + \frac{(c_y - j)^2}{b} \leq 1\ \}, \tag{6.36}$$

from which we compute the sample mean of square-intensities of cell pixels

$$\bar{C_p} = \frac{\sum_l {C_p^l}^2}{|C_p|} \tag{6.37}$$

To discriminate cells from background, the resultant mean square image is classified to cell and background by minimizing the inter-class variance as can be observed in Fig. 6.30(b).

Cell Template Generation

In contrast with the proposed mathematical cell template in Sec. 6.2 that was introduced based on attributes of a specific HSC phenotype such as uniform bright boundary and dark interior, and the proposed mathematical cell template in Sec. 6.4, that was introduced based on attributes of HSCs of different phenotypes, here using user interactions a more general cell template applicable to different cell types will be generated. In the proposed method, a user selects some cells in a few frames of the video clip by clicking on the upper-left and lower-right corners of a rectangular box that surrounds the cell. The selected cells are averaged to generate the cell template:

$$M_{tpl} = \frac{1}{N_c} \sum_{n=1}^{N_c} M_{rect}^n \tag{6.38}$$

where M_{tpl} and M_{rect}^n are 2-Dimensional matrices (masks). Then the cell template is convolved with the original image frame to generate a correlation map. Cell template and

correlation map obtained by applying the cell template are depicted in Fig. 6.31 and 6.32(a) respectively. The brighter pixels in the correlation map show the highly correlated points which are more likely to be a cell centre. To remove the unlikely cell centre candidates the correlation map is thresholded as depicted in Fig. 6.32(b).

Ellipse Fitting

So far we have segmented the cell regions using classification of elliptical mean square map to cell and background regions, and located the cell centre candidates by applying cell template and thresholding the correlation map. The set of cell parameters in a typical video frame is defined by f:

$$f = \{(c_x, c_y, a, b, \theta)|\ (c_x, c_y) \in C^c_{cand}\} \tag{6.39}$$

where (c_x, c_y) are cell centre coordinates which will be extracted from the set of cell centre candidates C^c_{cand}, (a, b) are the radii of the elliptical cell and θ is the orientation of the cell. To locate the cell centres, we propose the following *Maximum A Posteriori* (MAP) problem to be solved:

$$\hat{f} = arg\ \left\{ \underset{f}{max}\ \ P(f|\ I) \right\} \tag{6.40}$$

As it was already mentioned in this section, HSCs in our problem have a fairly circular shape, so to specialize the proposed elliptical model for HSCs, we set $a = b = r$ which results a circular shape and θ can be removed from the equation. Thus (6.39) can be simplified to:

$$f_{HSC} = \{(c_x, c_y, r)|\ (c_x, c_y) \in C^c_{cand}\} \tag{6.41}$$

and as a result (6.40) can be rewritten as

$$\hat{f}_{HSC} = arg\ \left\{ \underset{f_{HSC}}{max}\ \ P(f_{HSC}|\ I) \right\} \tag{6.42}$$

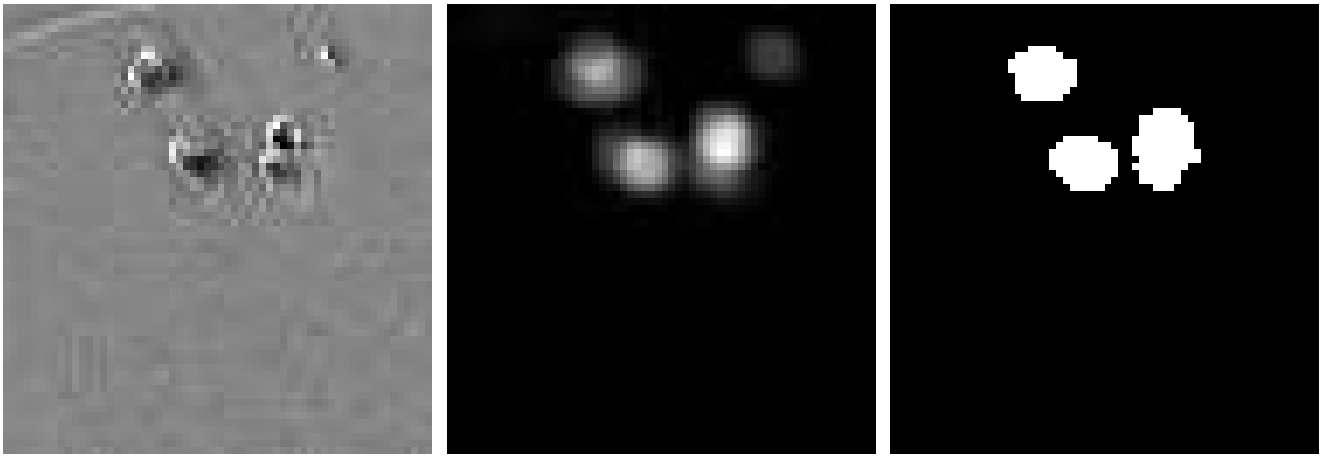
To solve the MAP problem (6.42) and find the cell centre and ellipse axis for each cell, we apply an optimized search method by fitting elliptical shapes to the segmented cell regions and searching for the maximum of $R(\alpha, \beta, \delta)$ defined as

$$R(\alpha, \beta, \gamma) = \frac{\alpha}{\beta + \gamma} \tag{6.43}$$

where α is covered area of the segmented region by ellipse (or ellipses), β is the area of the segmented cell region that is not covered by any ellipse, and γ is the area outside of the segmented cell region that is covered by an ellipse (or ellipses). The maximum of (6.43) must be found over the search space which consists of cell centre candidates C^c_{cand} and cell radius candidates C^r_{cand}. The former is obtained by locating all 1-Dimensional local maxima with distance one pixel in each row and each column of thresholded correlation map as it is depicted in Fig. 6.32(c). The latter is a $2 \times N_r$ matrix which is set empirically by observing HSC over different video clips, where N_r is the number of ellipse radius pairs.

6.5.2 Results

Figures 6.30 to 6.36 show the application of the proposed deconvolution method to a HSC image Phenotype 3 for cell localization. As can be observed the proposed deconvolution method is able to identify both non-dividing and dividing cell centers correctly. Figures show a visual representation of different stages of the proposed method applied to the HSC image Phenotype 3.



(a) (b) (c)

Figure 6.30: (a) Original HSC image. (b) Elliptical mean square of (a). (c) Binary image of (b) showing cell regions.

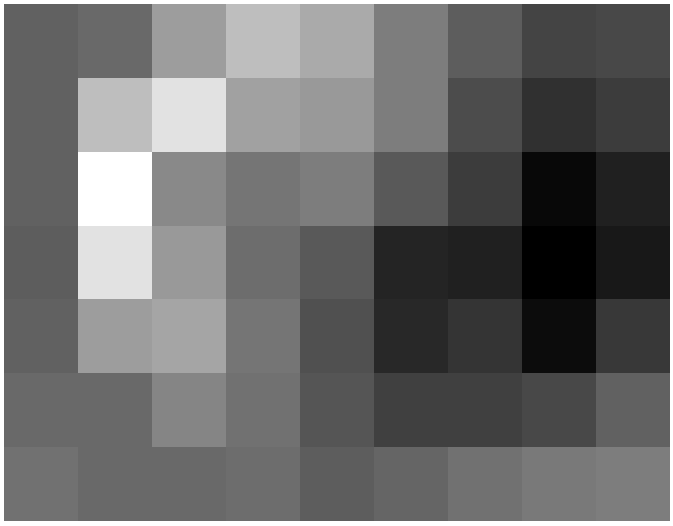
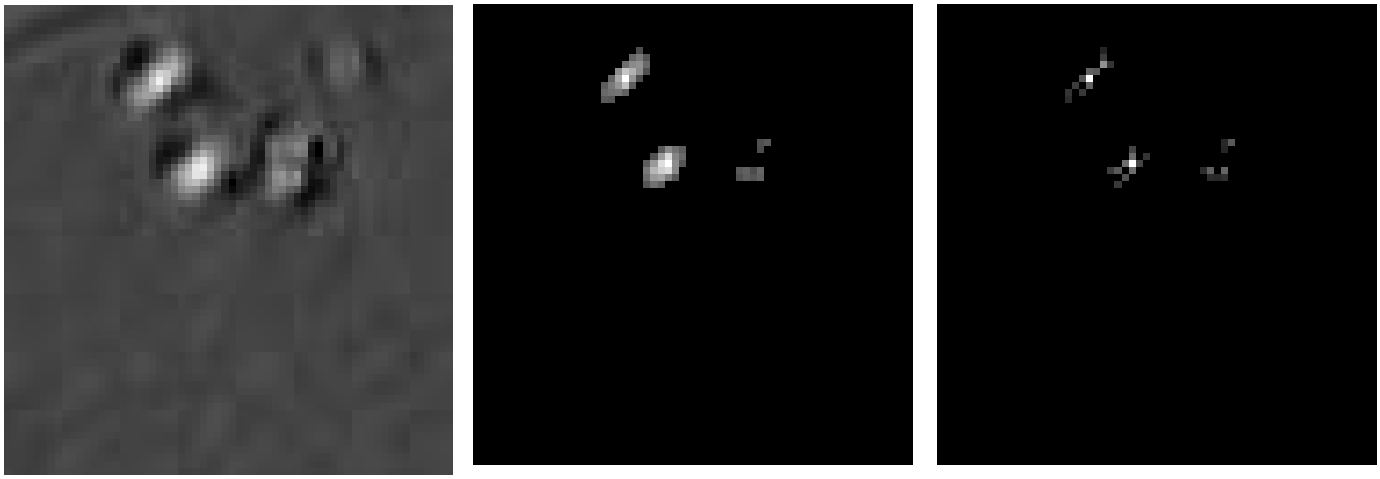
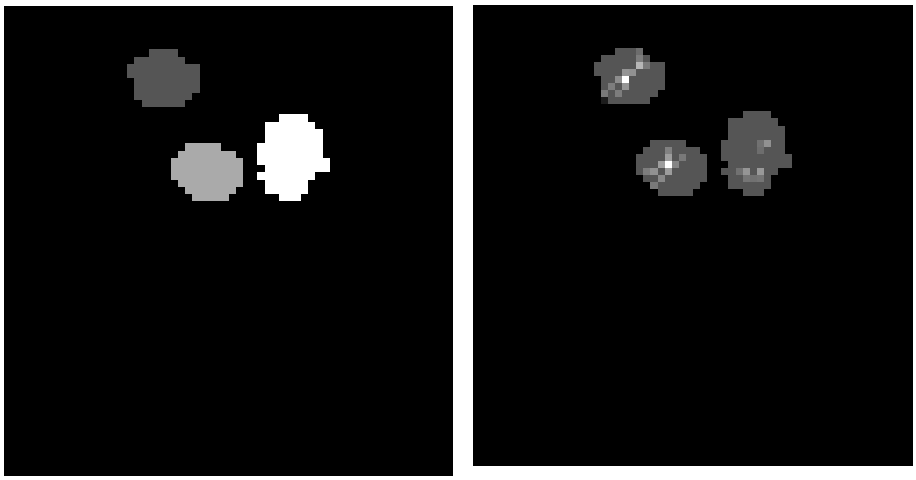


Figure 6.31: Sample cell template.



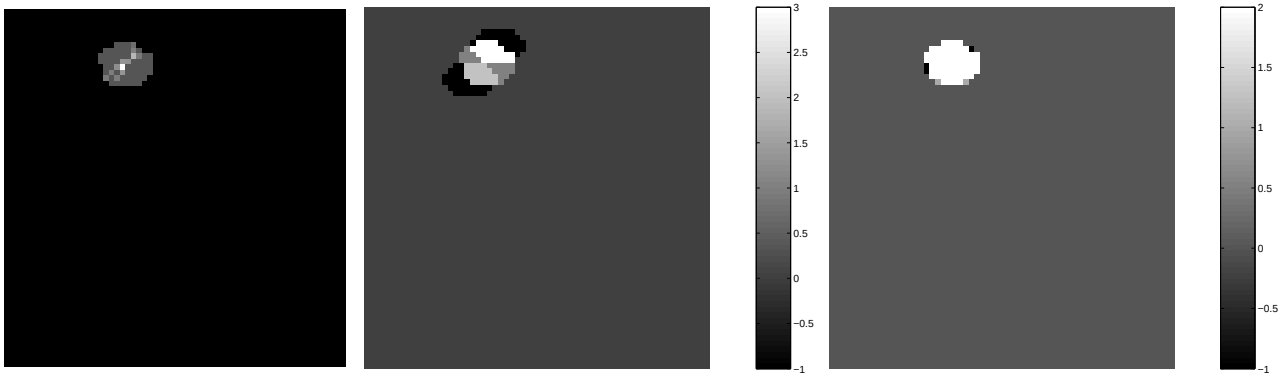
(a) (b) (c)

Figure 6.32: (a) Correlation map obtained by applying the cell template to the original image. (b) Thresholded correlation map. (c) All 1-D local maxima with distance 1-pixel located in each row and each column of (b).



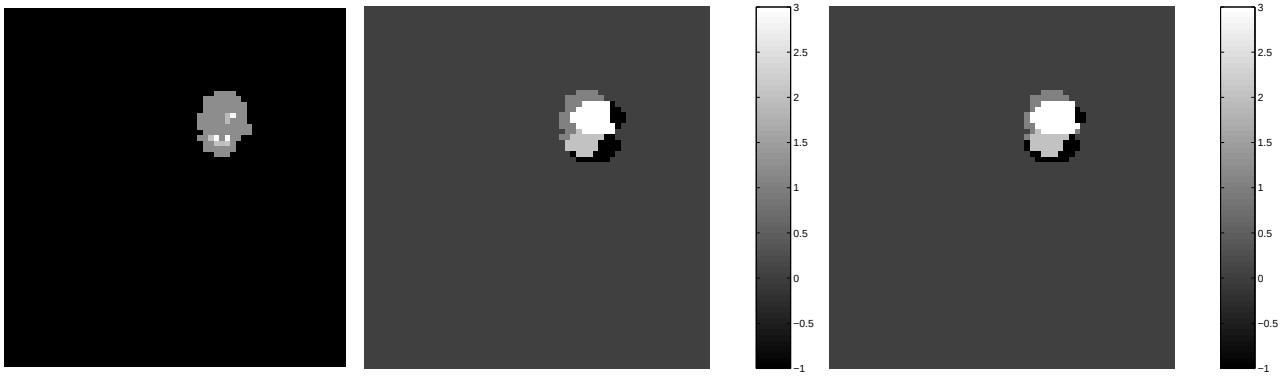
(a) (b)

Figure 6.33: (a) Labelling the cell segmented regions. (b) Superimposing local maxima in (Fig. 6.32(c)) on (a).



(a) (b) (c)

Figure 6.34: (a) K^{th} segmented region for $k = 1$ with superimposed cell center candidates. (b) A typical hypothesis $f_h^{Rig_k}$, fitting two elliptical cells in the segmented cell region ($k = 1$). (c) Optimal hypothesis $f_{best}^{Rig_k}$ which maximizes R for region $k = 1$.



(a) (b) (c)

Figure 6.35: (a) K^{th} segmented region for $k = 3$ with superimposed cell center candidates. (b) A typical hypothesis $f_h^{Rig_k}$, fitting two elliptical cells in the segmented cell region ($k = 3$). (c) Optimal hypothesis $f_{best}^{Rig_k}$ which maximizes R for region $k = 3$.

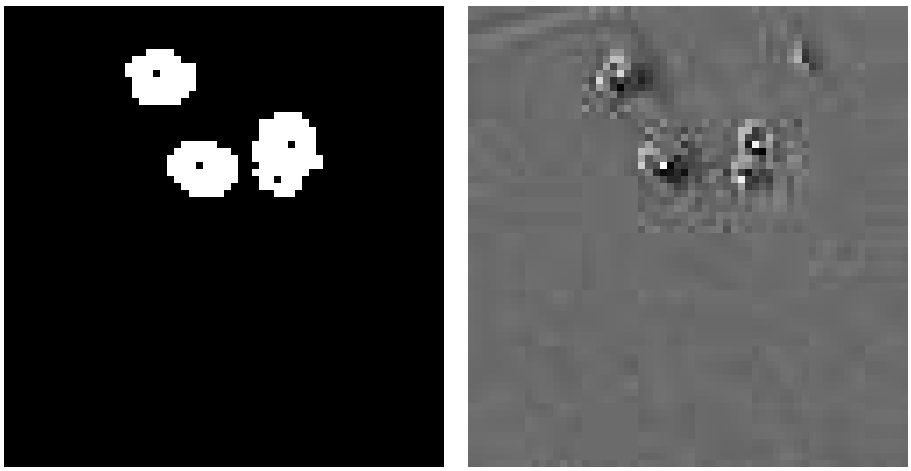


Figure 6.36: Located cell centers; Left: Superimposed on segmented HSC image. Right: Superimposed on original HSC image.

6.6 Watershed-Based Deconvolution for Cell Detection

In Sec. 6.5 we show that locating the cell centres is essentially an inverse problem which can be addressed in the form of a deconvolution problem. To solve the problem, we proposed to find a set of cell shape parameters for optimal representation of cell segmented areas. Thus the inverse problem was solved using an optimized ellipse fitting method and considering each ellipse centroid as a cell center. The proposed method effectively models dividing and close by cells, and is capable of modelling different cell types with changes in the model parameters. However in cases where either a complex parameterized shape is needed to model a cell, or an exact cell segmentation is in demand in place of cell center localization, this method will not be effective.

6.6.1 The Proposed Method

In this section we propose a method to achieve exact cell segmentation. Considering cell segmentation as an inverse problem, here again we address the solution in the form of a deconvolution problem. The key differences between this method and the method presented in Sec. 6.5 is that in Sec. 6.5

- We assumed that cell areas containing individual or groups of cells are segmented in advance.
- Then, the cell center localization was solved by finding a set of cell shape parameters for optimal representation of cell segmented areas.

In contrast, in the proposed method in this section

- We assume that cell centers are located in advance.

- Then, the cell segmentation will be solved by finding cell regions for optimal representation of cell centers.

The proposed method consists of cell template generation, template matching, cell center localization, and watershed segmentation.

Template Matching

Recall (6.38)

$$M_{tpl} = \frac{1}{N_c} \sum_{n=1}^{N_c} M_{rect}^{n} \tag{6.44}$$

A cell template is generated in the same way that was discussed in Sec. 6.5. The cell template, original image frame, and correlation map are depicted in Fig. 6.40, 6.41 and 6.42 respectively.

The brighter pixels in the correlation map show the highly correlated points which are more likely to be a cell centre. To locate the cell centres, we find the local maxima in the correlation map and thereshold the local maxima map as depicted in Fig. 6.43.

Edge Detection

An edge image as depicted in Fig. 6.44 will be obtained applying an edge detection method such as Sobel kernel to the original image. Cell areas then are coarsely segmented using morphological operators. This is accomplished by dilating a disk, morphological mask, over the edge image

$$G \oplus M \equiv \{G_i + M_j : G_i \in G,\ M_j \in M\} = \bigcup_{M_j \in M} G_{+M_j} \tag{6.45}$$

where G is the edge image, M is the mask, and

$$G_{+M_j} \equiv \{G_i + M_j :\ G_i \in G\} \tag{6.46}$$

is the translation of G along the M_j. Very small regions that are not likely to be a cell region will be removed after dilation. Coarse segmented image is depicted in Fig. 6.45.

Watershed Segmentation

As can be observed in Fig. 6.46, for the exact segmentation of cell boundaries, cell centers obtained by template matching are superimposed on a coarsely segmented binary image as local minima. The watershed method, as a hybrid region/boundary based segmentation approach, will then be used to partition the coarse segmented image by filling up the local minima, superimposed located cell centers, into disjoint homogeneous regions associated to individual cells. Therefore catchment basins represent cell regions, whereas watershed lines demonstrate the cell boundaries.

Let $I_s = (D, g)$ be the coarsely segmented image, where D is a digital grid and function $g : D \longrightarrow N$ assigns an integer value to each $d \in D$, and $g(d)$ is gray level of d corresponds to altitude in topographic context. Let $M_c = \{m_t |\ t \in [1, T_{M_c}]\}$ be the set of cell centers consisting of T_{M_c} centers which are superimposed as local minima on coarse segmented image. The set of points $d \in D$ which are topographically closer to a cell center m_i than to any other cell center m_j construct the cell region $C(m_i)$ which is associated with the cell center m_i.

$$C(m_i) = \Big\{ d \in D |\ \forall j \in I \backslash \{i\} :\ g(m_i) + S(d, m_i) < g(m_j) + S(d, m_j) \Big\} \tag{6.47}$$

where $S(d, m_t)$ is the topographical distance between d and m_t. The watershed of g is a set of points which do not belong to any cell region and represents the cell boundaries W:

$$W(g) = D \setminus \big(\cup_{i \in [1, T_{M_c}]} C(m_i) \big) \tag{6.48}$$

The exact segmentation applying the watershed is depicted in Fig. 6.47. Fig. 6.48 shows

superimposed cell centers and cell boundaries on the original image. As we can observe, cell boundaries are perfectly segmented for individual and dividing cells.

6.6.2 Results

Figures 6.40 to 6.48 show the application of the proposed watershed deconvolution method to a typical cell image for cell segmentation. As can be observed the proposed watershed method is able to identify both non-dividing and dividing cell regions correctly. Figures show a visual representation of different stages of the proposed method.

6.7 Discussion

The proposed cell detection methods in this chapter are compared and summarized in the Tabs. 6.37 and 6.38. Tab. 6.37 shows the applicability of the proposed cell detection methods to different cell data while the advantages of these methods are depicted in Tab. 6.38. As we can see, for example the initial cell model in Sec. 6.2 performs equally well as the thresholding detection method in Sec. 6.3 and general probabilistic model in Sec. 6.4, however this model is specifically designed based on attributes of regular HSCs and performs very poorly where is applied to the other cell data.

Cell Data / Applicability of Cell Model	Sec. 6.2	Sec. 6.3	Sec. 6.4	Sec. 6.5	Sec. 6.6
Regular HSCs	Yes	Yes	Yes	Yes	Yes
Bright HSCs	No	Yes	Yes	Yes	Yes
Poor Contrast HSCs	No	Yes	Yes	Yes	Yes
New High Resolution HSCs	No	No	No	No	Yes

Figure 6.37: The applicability of the proposed cell detection methods to different data sets.

The deconvolution model in Sec. 6.5 does a better job for modelling dividing cells but its cost is significantly higher than the probabilistic model in Sec. 6.6. However the deconvolution model is a generic approach for cell localization and can be adapted to detect/segment different cell data in cellular images captured using different imaging techniques. In Sec. 6.6, the deconvolution approach is specialized for exact cell boundary segmentation in high resolution cell data.

Advantages / Cell Model	Sec. 6.2	Sec. 6.3	Sec. 6.4	Sec. 6.5	Sec. 6.6
Generality	Specific	General	General	General	Specific
Complexity	$\bigcirc(n^2)$	$\bigcirc(n^2)$	$\bigcirc(n^2)$	$\bigcirc(c^n)$	$\bigcirc(n^2)$

Figure 6.38: Attributes and advantages of different detection method.

The effect of additive noise on cell detection is depicted in Tab. 6.39. Although additive noise does not have significant effects on mis-detection rate, it has direct relation with the false alarm rate. As can be seen in Tab. 6.39 the false alarm rate increases significantly by decreasing the *PSNR* below 14.

PSNR	Inf	16.1	14.5	14	13.5	12.4	11.8
False Alarm Rate	$< 7\%$	$< 7\%$	$< 10\%$	$< 20\%$	$< 35\%$	$> 50\%$	$> 60\%$
Mis-Detection Rate	$< 10\%$	$< 10\%$	$< 10\%$	$< 10\%$	$< 10\%$	$< 10\%$	$< 10\%$

Figure 6.39: The effect of noise in false alarm rate.

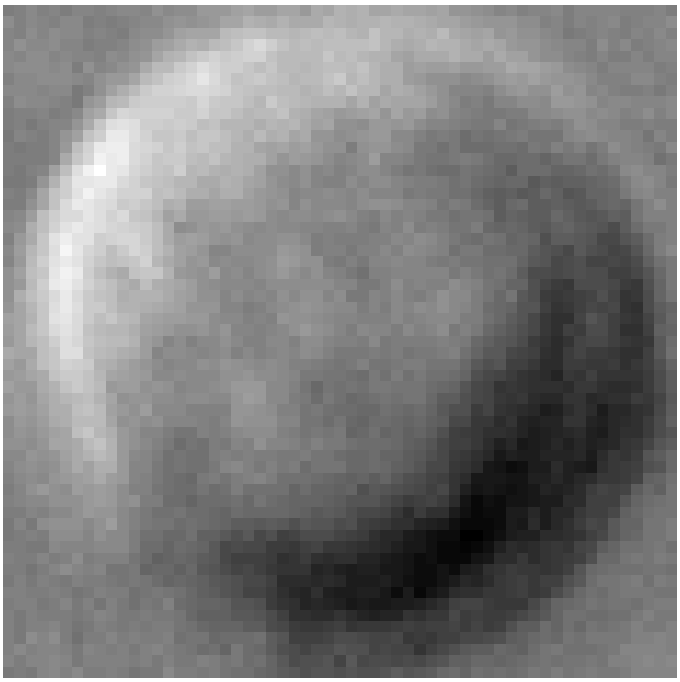


Figure 6.40: Cell template.

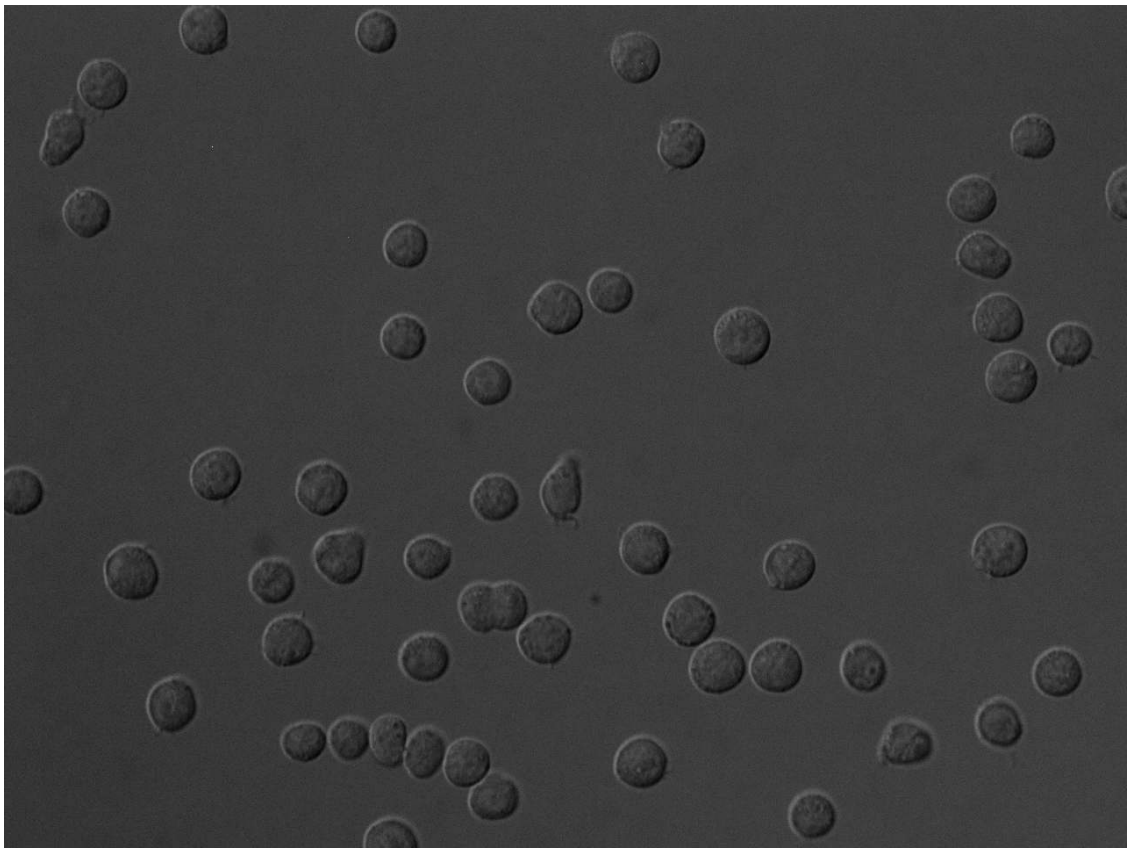


Figure 6.41: Original image.

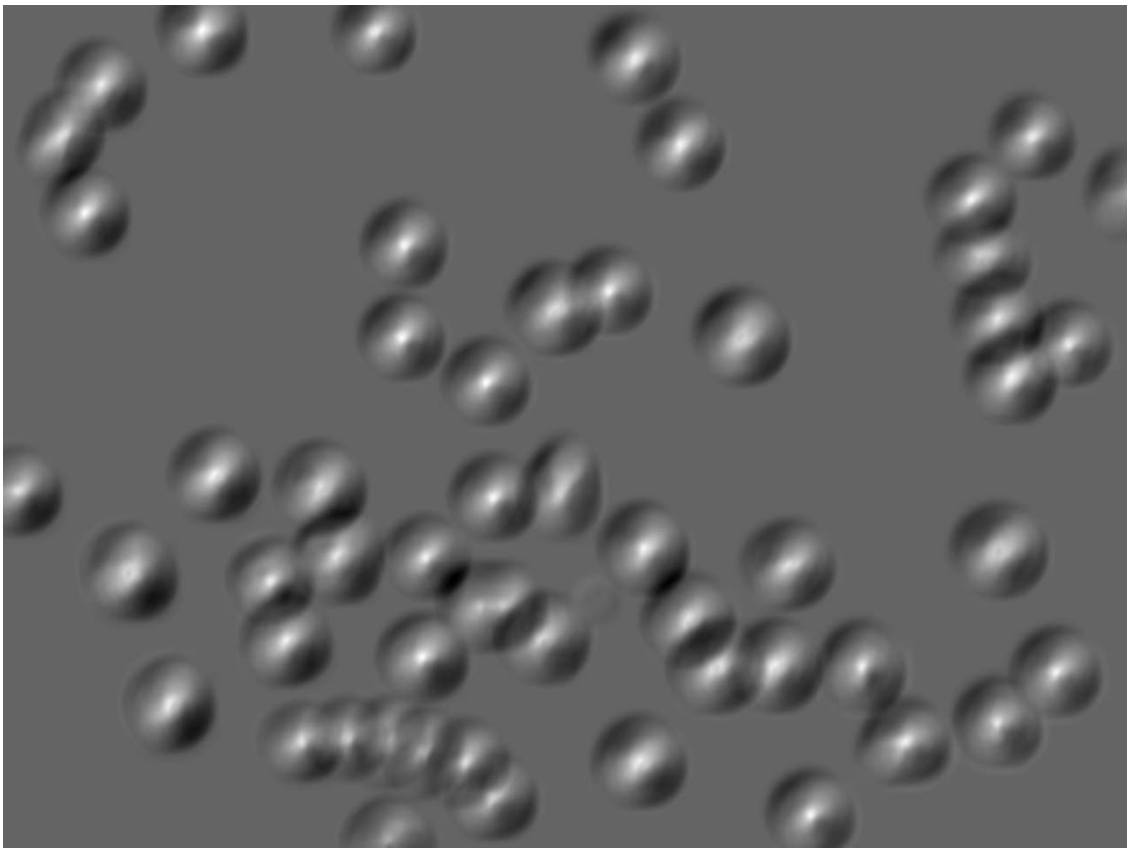


Figure 6.42: Correlation map that is obtained by convolving the cell template (Fig. 6.40) and original image (Fig. 6.41).

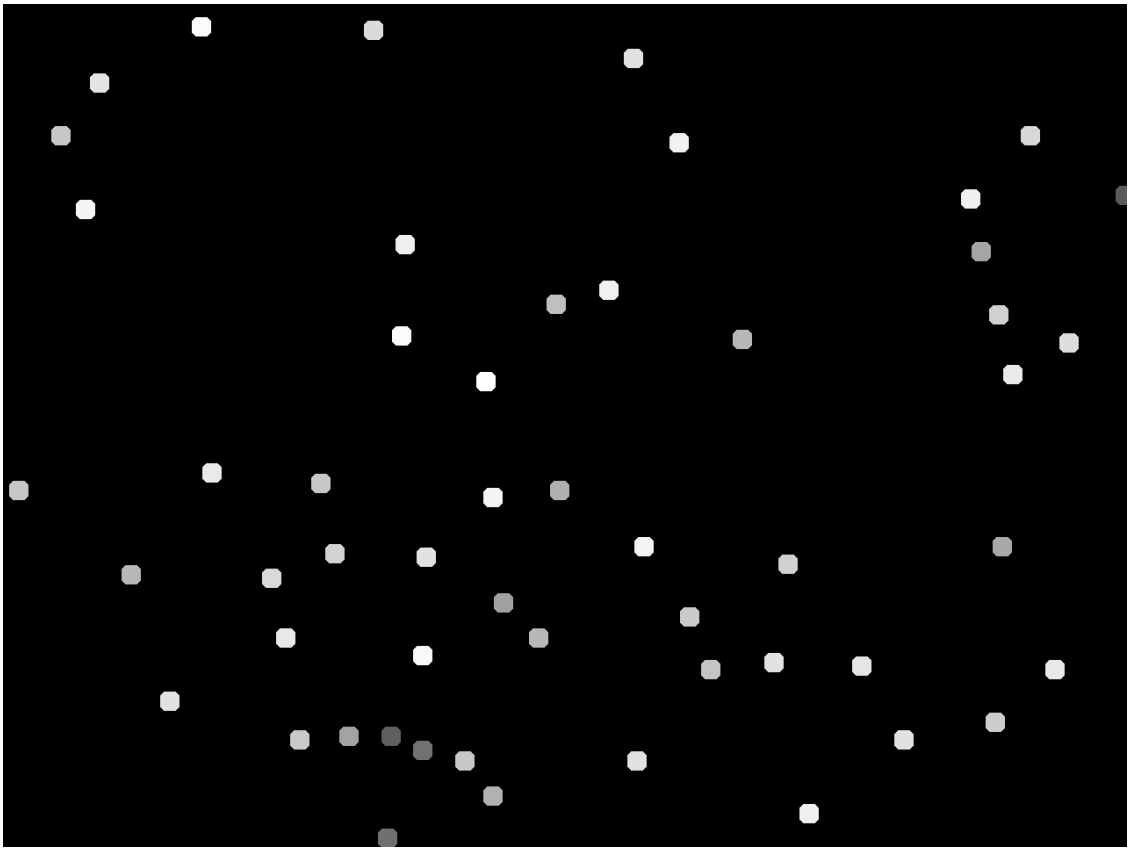


Figure 6.43: Located cell centers obtained by finding local maxima in (Fig. 6.42) and thresholding the local maxima map.

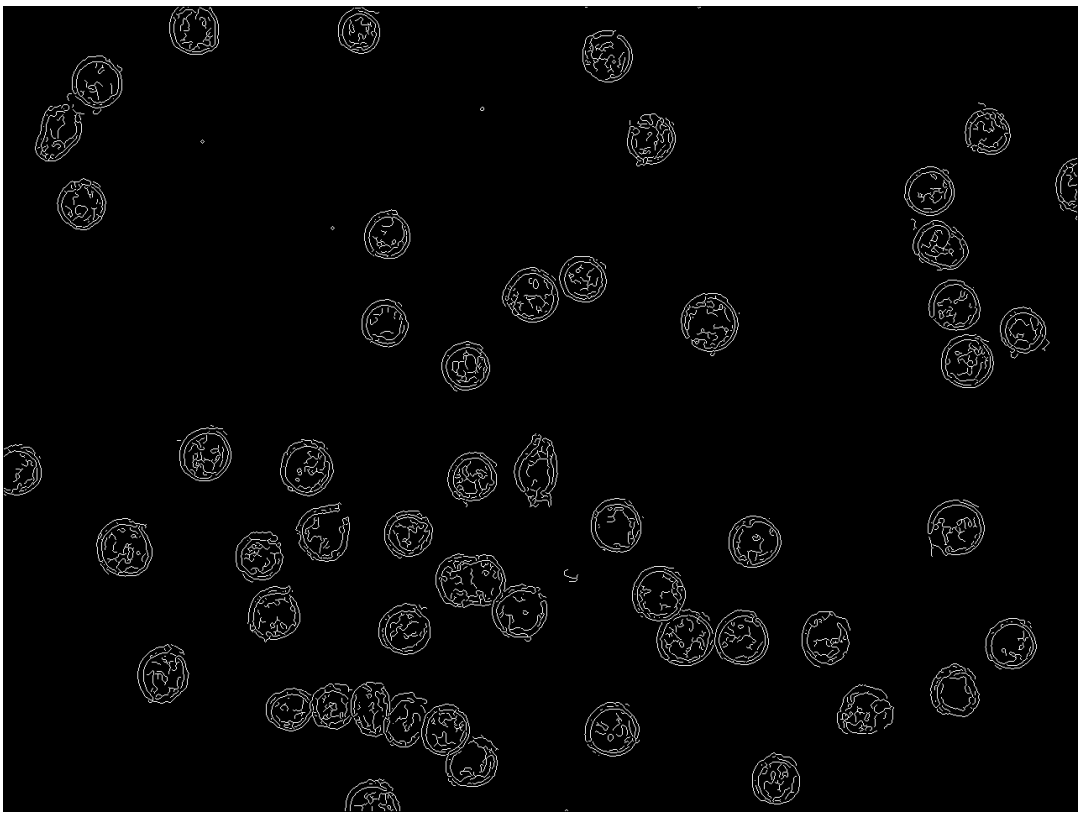


Figure 6.44: Edge detection applying the Sobel kernel to the original image in (Fig. 6.41).

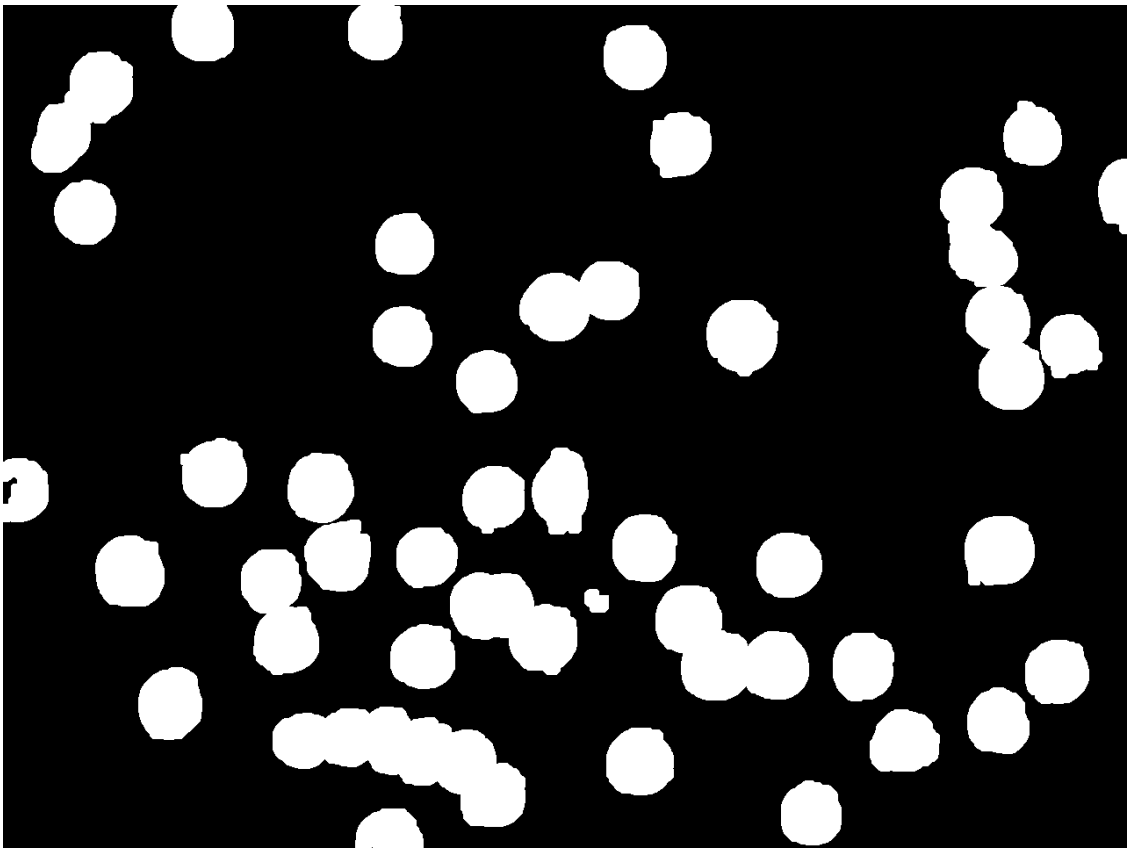


Figure 6.45: Dilating the edge image in (Fig. 6.44) by a disk.

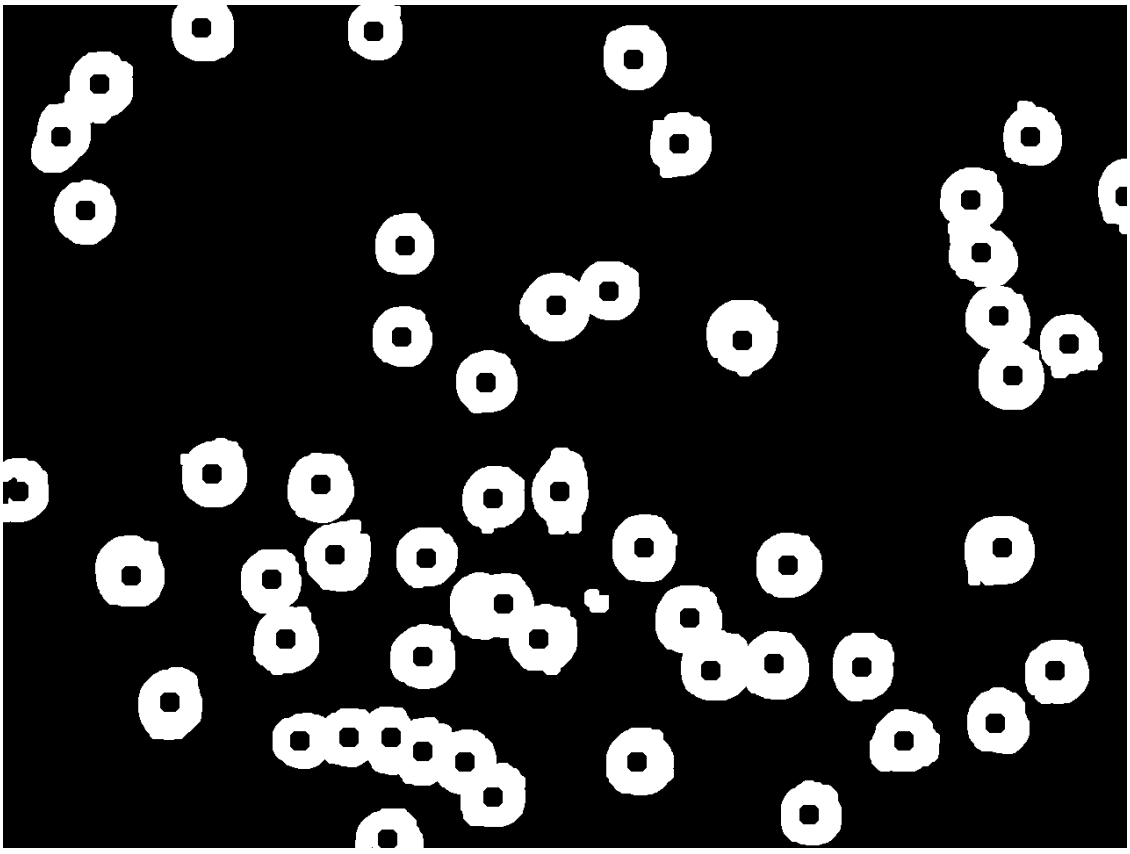


Figure 6.46: Superimposing the cell centers in (Fig. 6.43) on the dilated image in (Fig. 6.44) as local minima.

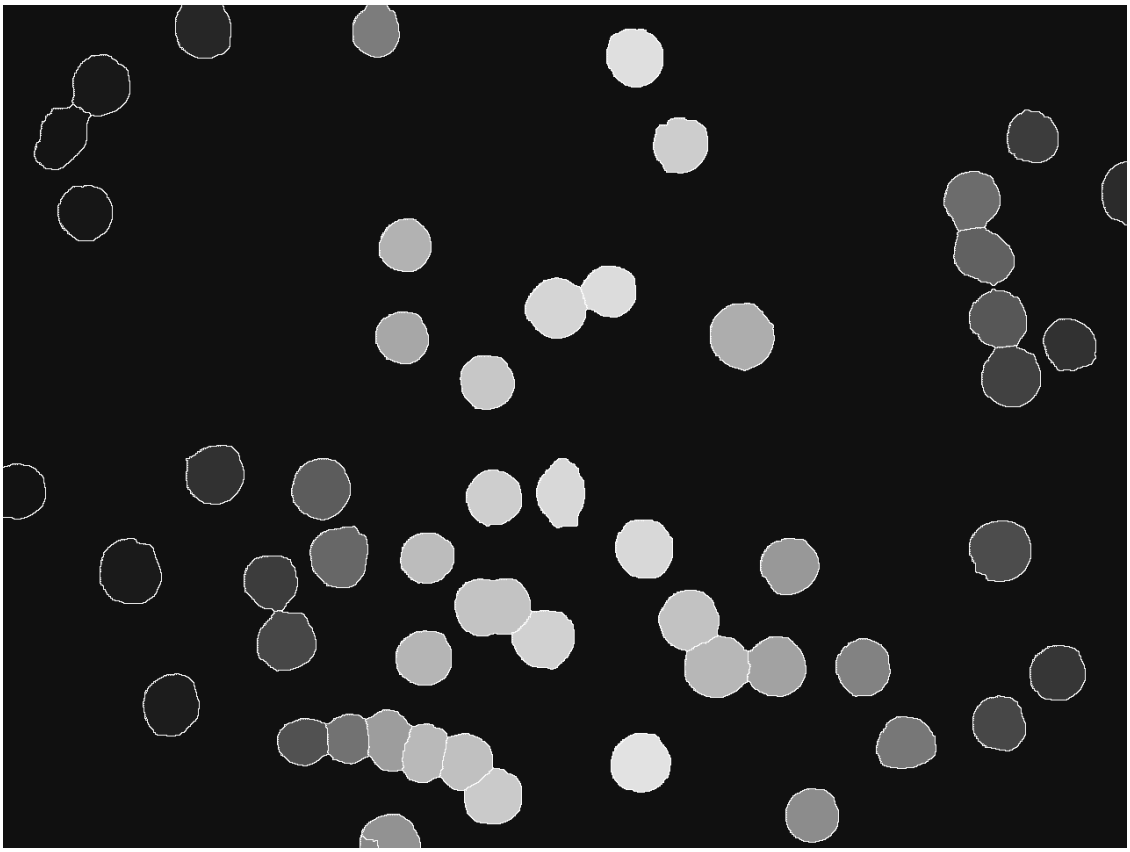


Figure 6.47: Segmented cell regions applying the watershed segmentation to (Fig. 6.46).

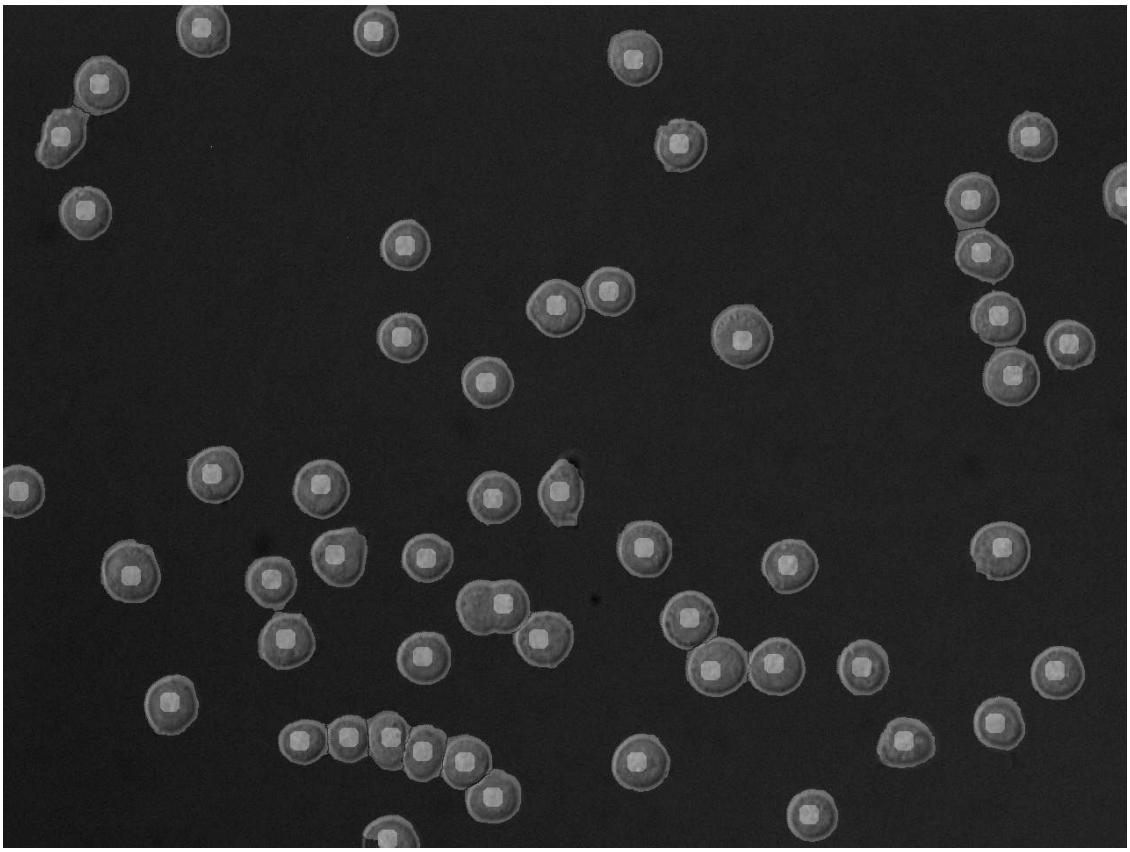
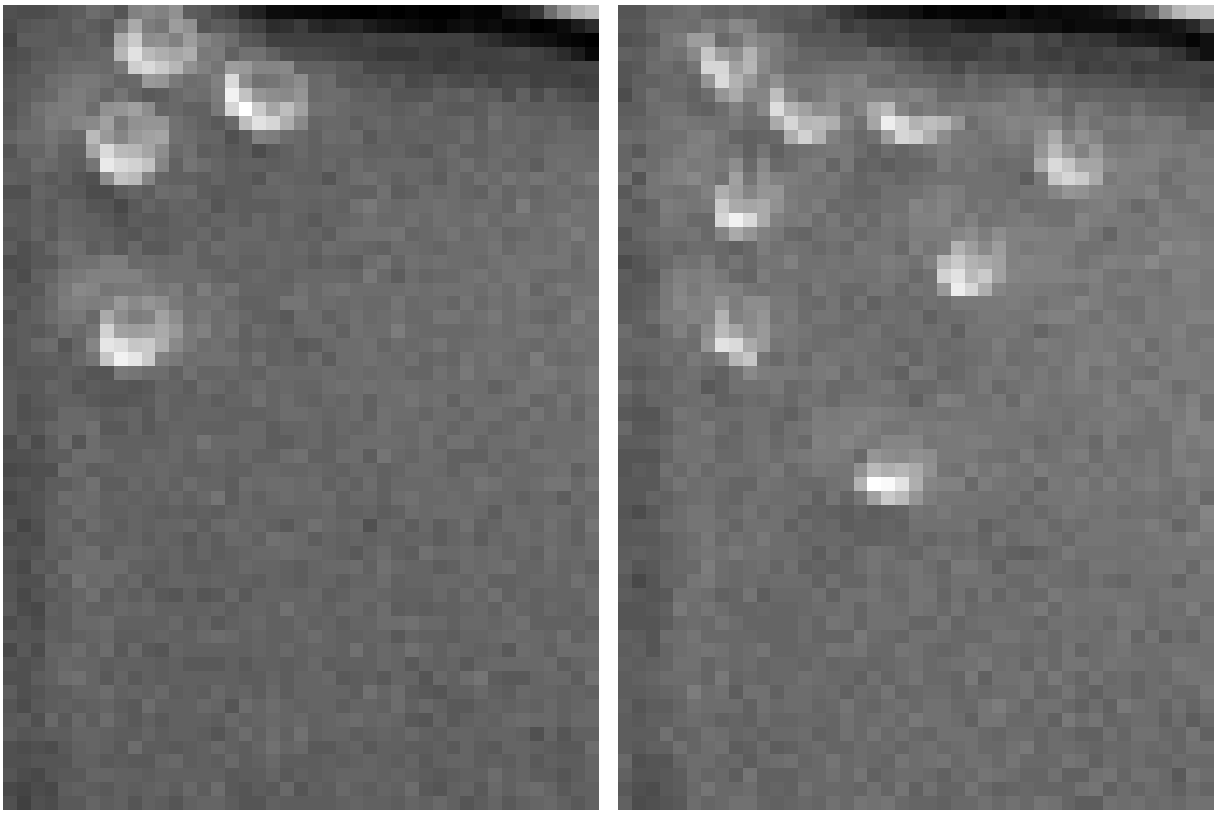


Figure 6.48: Segmented cell boundaries in (Fig. 6.47) and cell centers in (Fig. 6.43) are superimposed on the original image in (Fig. 6.41).

Chapter 7

Background Estimation

The previous chapter discussed cell detection. In this chapter the second important part of our cell tracking system, background estimation, will be introduced and discussed. Background estimation is one of the most challenging problems in tracking applications. The precision of the background estimation method usually has direct impact on the performance of segmentation/detection, as is the case in this research, that is cell tracking. Hence an accurate background estimation can indirectly increase the performance of tracking system. Our interest is an accurate estimation of the background scene where our image sequences are distorted both temporally and spatially because of illumination variations, spatial non-uniformities, additive noise, and stationary objects such as well boundaries. To improve the cell detection performance, an effective two-pass background estimation method is proposed in this chapter. In the first pass stationary objects such as debris and well boundaries are removed to improve the segmentation result. In turn in the second pass, the segmentation result is used to remove the foreground objects (cells) and improve the estimation result.



(Frame 1) (Frame 50)

Figure 7.1: Two unprocessed blood stem cell images of Phenotype 1.

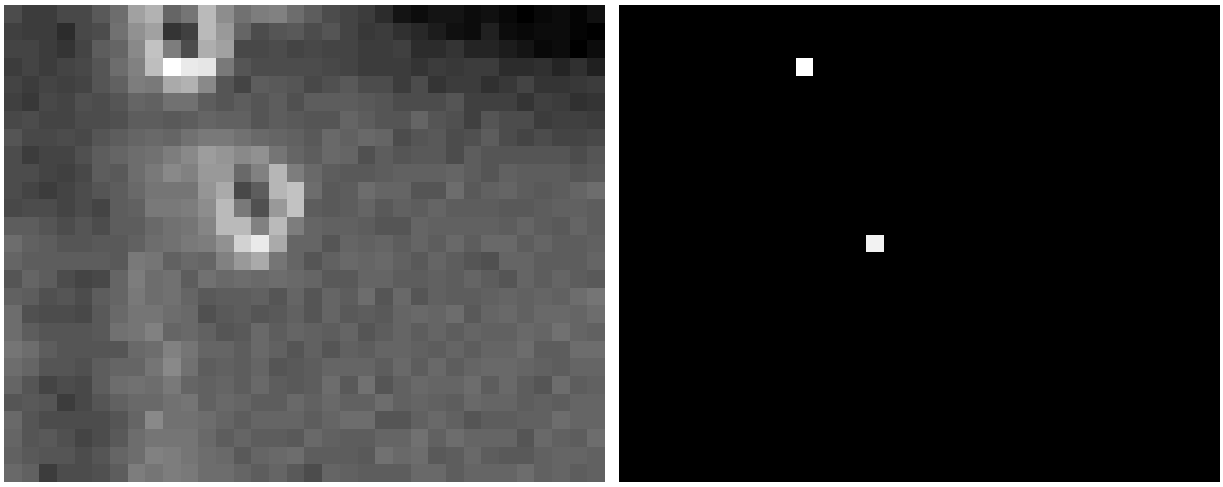
7.1 Background Estimation for Biomedical Applications

There are a broad range of biomedical applications, each of which introducing a different method to estimate the background based on some specific assumptions relevant to the problem [16, 26, 29]. Close and Whiting [29] introduced a technique for motion compensated estimation of background structure and artery in coronary angiogram images to distinguish the artery and background contributions to the intensity. They modelled the image in a region of interest as sum of two independently moving layers, one consist of the background structure and the other consist of the artery. The density of each layer varies only by rigid translation from frame to frame and the sum of two densities is equal to the image density.

Boutenko *et. al* [16] assumed that the structures of interest are darker than the surrounding immobile background and used a velocity based segmentation to discriminate vessels and background in X-ray cardio-angiography images considering the faster vessel motion in comparison with the background motion.

Chen *et. al* [26] modelled the background of a given region of interest using the temporal dynamics of its pixels in quantitative fluorescence imaging of bulk stained tissue. They modelled the intensity dynamics of individual pixels of a region of interest and derived a statistical algorithm to minimize background and noise to decompose the fluorescent intensity of each pixel to background and the stained tissue contributions.

In the rest of this chapter, we describe the problem at hand first. Then a method is proposed for mutual foreground segmentation and background estimation.



(a) (b)

Figure 7.2: (a) Cropped well interior without well boundaries. (b) Applying the cell model in (6.2), to the cropped well interior.

7.2 Problem Description

Frames 1 and 50 of a cropped well are depicted in Fig. 7.1. A typical microscopic multi-well image sequence in our experiments consists of 32 independent wells, in each of which two to four HSCs are injected. Hence single-well image sequences are cropped from the original multi-well image sequence and are processed individually. The first frame of a cropped well interior without well boundaries and the resultant detection after applying cell model (6.2) are shown in Fig. 7.2. The well cropping is often coarse and the well boundaries may be partially or completely visible in the cropped image sequence, as can be seen in Fig. 7.3(a). Applying the cell model in Sec. 6.2 to these coarsely cropped wells produces very poor results as can be observed in Fig. 7.3(b).

As the performance of the tracking algorithm relies closely on the quality of cell detec-

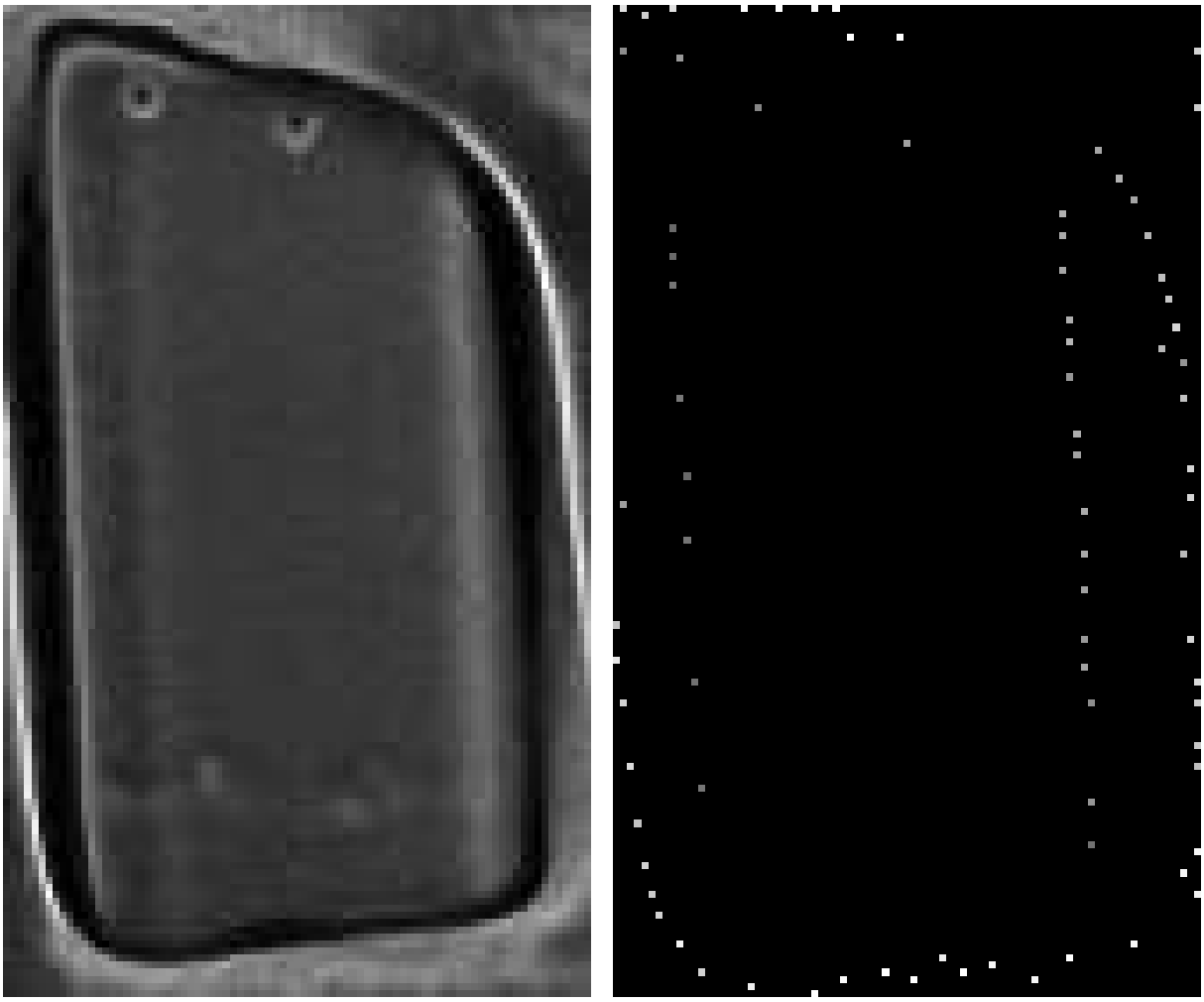
tion, a poor detection degrades the tracking performance. To obtain a uniform background, improve the signal to noise ratio and in turn the cell detection performance, the background intensity variations which have been caused by illumination variations and noise must be eliminated. To do this, the background image will be estimated and then subtracted from the original image sequence.

To detect and localize HSCs applying the proposed cell models in chapter 6, we assumed that the image sequence has a uniform background. Hence, the background estimation/subtraction algorithm (that will be discussed in this chapter) will be applied to the image sequences without a uniform background before the application of a cell model. The background estimation/subtraction method is discussed in this chapter.

We are interested in the tracking of Hematopoietic Stem Cells (HSCs) in culture to analyze stem-cell behavior and infer cell features. The limited cell variability and the rigid camera mount lead to a number of key differences between our problem and most video sequences in computer vision, summarized in Tab. 7.1.

Most tracking problems have an implicit, nonparametric model of the background to avoid making assumptions regarding the foreground. By developing a model for the background it is possible to find a classifier that labels each image pixel as background/not background; i.e., the foreground is identified as that which is not background. In contrast, the more focussed context of our cell tracking problem admits an explicit model of the foreground. Because of the low SNR of our problem, where illumination is limited to minimize cell phototoxicity, it is desired to remove all deterministic non-cell variations in the image (i.e., the background) before localizing the cells.

Moreover, in many computer vision problems, such as vehicle tracking, a fixed vehicle is intended to be part of the background, therefore the preponderant statistics for any pixel may be considered to be background. On the other hand, a nearly stationary cell,



(a) (b)

Figure 7.3: (a) Coarsely cropped well in which well boundaries are visible. (b) Applying the cell model in (6.2), to the coarsely cropped well. In contrast with the perfect result obtained by applying the cell model in (6.2) to a cropped well interior without boundaries (Fig. 7.2(b)), here the cell model (6.2) performs very poorly where the cropped well contains visible well boundaries.

Table 7.1: Our background estimation problem vs. conventional ones

Our background estimation problem	**Conventional background estimation**
Low signal to noise ratio (SNR) due to phototoxicity	High SNR
Rigid camera	Possible camera wobble
Static background	Dynamic background
Explicit foreground model	Parametric/nonparametric background model

as appears near the middle of Fig. 7.1(Frame 1), must not be considered background. Therefore, most of the present background models fail to estimate the background correctly in such spatial locations.

Although cell localization would appear to be a foreground/background classifier, there is a difference: we do not need to actually segment the image, only to identify the cell locations. Therefore, crucially, we do not need to reliably classify each pixel definitively as foreground or background, with unavoidable error around the cell margins. Rather to accurately estimate the background we only need to identify most background pixels, most of the time.

Next, whereas most tracking problems in computer vision need to accommodate camera jitter (due to vibration or wind, for example) and possibly a dynamically changing background, many problems of scientific imaging have a rigidly mounted camera and a static background. Thus, by using a model to find the foreground, we have a rational means of identifying portions of background, from which a static background can be estimated. In most previous work, background modelling has been employed to classify image pixels to foreground and background, however to the author's knowledge none of the earlier works has integrated foreground detection and background estimation in a mutual frame-

work. Localized cells, the foreground elements, are first removed from the scene frame by frame. The resultant residual image sequence is then used for an accurate spatio-temporal background estimation.

7.3 The Proposed Background Estimation Method

We address background estimation issues and propose a two-pass estimation-detection method to precisely estimate the background using detected foreground objects. As opposed to the more common approach of detecting foreground anomalies given a background model, a foreground model is applied for cell detection in a uniform background.

The proposed cell model in Sec. 6.4 is applied to an image sequence to find cell candidates. The segmented cell regions are removed from the image sequence, and the residual is then used for an accurate spatio-temporal background estimation. The proposed algorithm will be applied to estimate/detect the background/cells in phase contrast video taken from living Hematopoietic Stem Cells in culture.

The proposed background estimation method consists of two passes. The first pass estimates the background, which is then subtracted from the original image sequence, in turn improving the quality of cell detection when the proposed cell model is applied to the corrected image sequence. In the second pass a more precise estimate of the background is inferred by removing cell pixels, inferred by the cell model, from the original image sequence.

7.3.1 First Pass: Pointwise Background Estimation

To estimate the background, let $\mathbf{I} = (I_1, I_2, \cdots, I_K)$ be a set of K images, thus for each frame I_k of an image sequence we can write

$$I_k = F_k + B + n_k \cdot \mathit{1} + V_k \tag{7.1}$$

where F_k is the dynamic foreground, B is the fixed background, n_k models the temporal variations in global lighting, and V_k is spatio-temporal random additive noise. The temporal noise n_k is estimated over all pixels of the frame k

$$\hat{n}_k = \underset{\{ij\}}{mean}\ (I_{ijk})\ \ \{(i,j)|\ i \in [1,N]\ \&\ j \in [1,L]\} \tag{7.2}$$

For temporal correction, the estimated temporal noise is subtracted from the original frame I_k:

$$g_k = I_k - \hat{n}_k \cdot \mathit{1} \tag{7.3}$$

$B = [B_{ij}]$, composing of stationary distortions and illumination variations at each pixel location, is estimated over K frames of temporal corrected sequence $\mathbf{g} = (g_1, g_2, \cdots, g_K)$:

$$\hat{B}_{ij} = \underset{\{k\}}{mode}\ (g_{(i,j,k)})\ \ \{k \in [1,K]\} \tag{7.4}$$

and is subtracted out from $\mathbf{g}$:

$$\hat{F} = \mathbf{g} - \hat{B} \tag{7.5}$$

An imperfectly cropped well is depicted in Fig. 7.4(a), corresponding estimated background and corrected image, based on the first pass of the background estimation, are depicted in Figs. 7.4(c) and (d) respectively. Fig. 7.5 shows different HSC phenotypes, estimated backgrounds, and corrected frames after background subtraction.

Justification of mode as statistical measure

Empirically, the motion of blood stem cells is essentially random, especially when observed minutes apart. Since cell motion is rarely zero, the spatial variations in cell brightness mean that the variability of an image pixel, located within a cell, is considerably higher than the variability of an image pixel lieing in the background, whose variability is due only to random noise. Therefore, excepting cases of unusually small cell motion, the distribution of brightness values at a pixel should be most sharply peaked at the background, which is therefore recovered by the mode of the sample histogram.

7.3.2 Second Pass: Multi-Clique Background Estimation

As the foreground cells are essentially outliers relative to the background statistics, to precisely estimate the background, we need to identify and remove the foreground, thus we remove the pixels which are associated to the located cells in each frame, specifically all pixels inside a rectangular box with side length $2r_k^m$, centered at (x_k^m, y_k^m).

To remove the foreground elements, cell center locations are obtained by applying the cell model in (6.23) to the background subtracted images. Located cell centers are depicted in Fig. 7.6 for Phenotype 1. The images after removing the objects are depicted in Figs. 7.7(a) and 7.7(b) for frames 1 and 50 respectively. Because of imperfect segmentation, cell fragment pixels might appear in the residual image frames after removing the located cells from the original image frames. Depicted in Fig. 7.8, we can see the temporal intensity variations of a cell fragment pixel against that of a typical background pixel.

Recalling from (7.1), $I_k = F_k + B + n_k \cdot \mathbf{1} + V_k$, the cells are localized by applying ((6.23) from Sec. 6.4) to the corrected image sequence $\hat{F}$ from (7.5), generating a set of located cell centers

$$Z_k = \{z_k^m |\ m \in [1, M_k]\} \tag{7.6}$$

where M_k is the number of located cells in frame k.

A residual sequence is obtained by removing the localized cells frame by frame. Each residual frame R_k contains all pixels of original frame I_k except those belonging to the foreground cells (H_k) and which have been removed. The set of pixels to remove is

$$H_k = \{(p,q) \mid \ p \in [x_k^m - r_k^m, x_k^m + r_k^m], q \in [y_k^m - r_k^m, y_k^m + r_k^m], m \in [1, M_k]\} \tag{7.7}$$

Assuming all of the cells have been perfectly removed, each remaining pixel represents a noisy sample of the background:

$$R_{ijk} = \{B_{ij} + n_k + V_{ijk} \mid \ (i,j) \in L \setminus H_k\} \tag{7.8}$$

where L is the lattice of image pixels. The temporal noise n_k can be estimated as

$$\hat{n}_k = \underset{\{(i,j)\in L\setminus H_k\}}{mean} \ (R_{ijk}) \tag{7.9}$$

an improved estimate over that of (7.2), since now predominantly background pixels are used to compute the estimate, as opposed to the entire image in (7.2). For temporal correction, the estimated temporal noise is subtracted from the residual frame:

$$g_k = R_k - \hat{n}_k \cdot \mathbf{1} \tag{7.10}$$

The temporally corrected residual sequence $\mathbf{g} = [g_{ijk}]$ satisfies

$$\begin{aligned} g_{ijk} &= \{B_{ij} + \tilde{n}_k + V_{ijk}| \ \ (i,j) \in L \setminus H_k\} \\ &\simeq \{B_{ij} + V_{ijk} \mid \ (i,j) \in L \setminus H_k\} \end{aligned} \tag{7.11}$$

where $\tilde{n}$ is the temporal estimation error. We then precisely estimate the background $B = [B_{ij}]$, consisting of spatially stationary distortions and illumination variations at each pixel location over K frames of temporally corrected residual sequence $\mathbf{g} = (g_1, g_2, \cdots, g_K)$:

$$\hat{B}_{ij} = \underset{(p,q,r)\in Q(i,j,D)}{t\_mean^{\Omega}} \ (g_{pqr}) \tag{7.12}$$

where t_mean^{Ω} is the trimmed mean, with trimming parameter Ω, calculated by sorting the values $g_{p,q,r}$, removing the first and last $\Omega\%$, and computing the sample mean over the remaining samples. Application of the trimmed mean performs well by removing 10% from the first and last based on our experiments, so Ω is set to 10. The remaining question, addressed in the following section, is how to choose which elements Q to include in the trimmed mean.

Justification of trimmed mean as statistical measure

In principle the earlier justification of mode applies equally to the second pass of background subtraction, however there are two factors that motivate a choice of trimmed mean:

1. In contrast to the first pass, where some pixels may be dominated by quasi-stationary cells, here the cell removal implies that a large majority of the remaining pixels should belong to the background, therefore a mean-like approach is appropriate.

2. The computation of mode can be highly noise-sensitive, in contrast to the robust and straightforward computation of means and trimmed-means.

Therefore we expect the intensities of a given pixel, over time, to equal the background intensity plus noise, with rare outliers introduced by miss-detected cells or by cell fragments due to imperfect segmentation. To reject these outliers a trimmed mean is proposed.

7.4 Neighbourhood Selection

Removing the located cells aggressively from the image sequence reduces the number of background samples in some spatial locations, possibly causing an inaccurate estimation of background as we can observe in Fig. 7.9. In order to improve the accuracy of background

estimation, it would be desirable to include as many pixels as possible in Q in (7.12), possibly including image pixels from a neighbourhood N of spatial locations near to the background pixel being estimated. Clearly there is a tradeoff here: as the neighbourhood expands the number of elements in Q grows, allowing for greater noise reduction in the computation of $\hat{B}$ in (7.12), but at the same time causing more and more spatial blurring. In this section we address the question of deriving an optimal neighbourhood choice, based on the spatial statistics of the background.

$Q(i, j, D)$ contains all pixels in the residual sequence which fall inside 3-Dimensional spatio-temporal neighbourhood N:

$$Q(i, j, D) = \{(p, q, r) \mid \ (p, q) \in N(i, j, D), \ (p, q) \in L \backslash H_r\} \tag{7.13}$$

By removing the located cells from the image sequence, we will have fewer samples for the spatial location of cell pixels leading to imprecise estimation of background in computing the trimmed mean.

To overcome this shortcoming, we design an $n \times n$ multi-clique neighbourhood, to allow the trimmed mean to be computed over a spatial extent, as illustrated in Fig. 7.10. Each clique has a specific orientation and extent, indexed from $0, 1, \ldots, 9 = T_c$. A neighbourhood $N(i, j, D)$, centered on location (i, j) is made up of the union of some number of cliques

$$N(i, j, D) = (i, j) \cup \Big\{p, q| \ c(i - p, j - q) = d, d \in D\Big\} \tag{7.14}$$

where

$$D \subseteq \{0, 1, \cdots, T_c\} \tag{7.15}$$

D can vary from pixel to pixel and selects a subset of the cliques to be used in computing the trimmed mean and will be determined by the spatial variance of samples in the multi-clique neighbourhood.

Let $q_{i,j,d}$ be the sample variance over $g_{N(i,j,d)}$, i.e., for each spatial location (i,j) the sample variance is computed using the pixel at (i,j) and neighbour pixels in clique d. We compute spatial variances

$$\left\{q_{i,j,0}, q_{i,j,1}, \cdots, q_{i,j,T}\right\} \tag{7.16}$$

where $q_{i,j,0} \equiv 0$. We then sort the spatial variances in ascending order

$$\left\{q_{i,j,d_1} \leq q_{i,j,d_2} \leq \ldots \leq q_{i,j,d_{T-1}}\right\} \tag{7.17}$$

where $d_1, d_2, \ldots$ are the sorted indices, in order to identify those cliques over which the background is smoothest, to find the most promising candidates for spatial averaging.

We wish to find the appropriate neighbourhood D over which to compute the trimmed mean in (7.12). Those cliques having the smallest spatial variance are the most promising for including in the trimmed mean, since the pixels in those cliques are most similar to the pixel being estimated. We therefore propose to construct a sequence of neighbourhoods $D_0, \ldots, D_T$, cumulatively incorporating the minimum-variance cliques, based on the sorted variances q_{i,j,d_t}:

$$\begin{aligned} D_0 &= \{0\} \\ D_1 &= \{0, d_1\} \\ D_2 &= \{0, d_1, d_2\} \\ &\cdots \\ D_T &= \{0, d_1, \cdots, d_T\} \end{aligned} \tag{7.18}$$

The remaining task, then, is to determine which of these neighbourhoods optimizes the estimation error variance of the estimated background. The optimizing neighbourhood will then be the one used in the trimmed mean.

To determine the error variance for each neighbourhood, recall (7.11):

$$\begin{aligned} g_{ijk} &= \{B_{ij} + \tilde{n}_k + V_{ijk} |\ \ (i,j) \in L \setminus H_k\} \\ &\simeq \{B_{ij} + V_{ijk} \mid\ \ (i,j) \in L \setminus H_k\} \end{aligned} \tag{7.19}$$

where $V_{ijk} \sim \sigma_w^2$. Because we removed the cell pixels (H_k) fairly aggressively, selecting a rectangular region to remove slightly larger than the cell radius, therefore the pixels remaining in $L \setminus H_k$ will be predominantly background pixels, with only the occasional outlier. Recall from (7.12) that we would like to use a robust trimmed mean

$$\hat{B}_{ij} = \underset{(p,q,r)\in Q(i,j,0)}{t\_mean^{\Omega}} \ (g_{pqr}) \tag{7.20}$$

to estimate the background. The error variance of this estimate is a function of the number of outliers, which cannot be predicted in advance, however assuming that all outliers are removed by trimming, and that the trimming fraction Ω is small, then the estimation error variance should be similar to that of the mean applied to outlier-free data:

$$\hat{B}_{ij} \sim \left(B_{ij}, \frac{\sigma_w^2}{\|Q(i,j,0)\|} \right) \tag{7.21}$$

where $\|\cdot\|$ counts the number of elements in the set. However where we have few background samples remaining at the spatial location (i,j), the background B_{ij} can be estimated over some spatial extent of pixels in a multi-clique neighbourhood to increase the number of samples. A neighbouring background value $B_{\alpha,\beta}$ is modelled as having a mean of $B_{i,j}$ with a variance of q,

$$B_{\alpha,\beta} \sim (B_{ij}, q_{i,j,d}) \tag{7.22}$$

where (α,β) is a neighbour of (i,j) and

$$d = c(i-\alpha, j-\beta) \tag{7.23}$$

Thus given a neighbourhood $N(i,j,D)$, where D is a set of accumulated cliques, we can compute an estimate of a single background pixel based on all of the measured pixels within the given neighbourhood

$$\hat{B}_{ij} = \frac{\sum_{(\alpha,\beta)\in N(i,j,D)} \left(\sum_{k\in Q(\alpha,\beta,D)} \ (g_{\alpha\beta k}) \right)}{\sum_{(\alpha,\beta)\in N(i,j,D)} \|Q(\alpha,\beta,0)\|} \tag{7.24}$$

That is, the estimate is essentially a weighted sum of estimates of neighbouring pixels, weighted by the number of available measurements:

$$\hat{B}_{ij} \ = \ \sum_{(\alpha,\beta)\in N(i,j,D)} \hat{B}_{\alpha\beta} \cdot \frac{\|Q(\alpha,\beta,0)\|}{\sum_{(\alpha,\beta)\in N(i,j,D)} \|Q(\alpha,\beta,0)\|} \tag{7.25}$$

However $\hat{B}_{\alpha\beta}$ is a random variable, of mean $B_{\alpha\beta}$ and variance $\sigma_w^2/\|Q\|$, therefore (7.25) can be expressed as

$$\hat{B}_{ij} \ = \ \sum_{(\alpha,\beta)\in N(i,j,D)} \left(B_{\alpha\beta}, \frac{\sigma_w^2}{\|Q(\alpha,\beta,0)\|} \right) \cdot \frac{\|Q(\alpha,\beta,0)\|}{\sum_{(\alpha,\beta)\in N(i,j,D)} \|Q(\alpha,\beta,0)\|} \tag{7.26}$$

From (7.22) we have a model of the spatial relationship of B_{ij} and $B_{\alpha\beta}$, which leads to the estimate

$$\hat{B}_{ij} \ \simeq \ \frac{\sum_{d\in D} \sum_{(\alpha,\beta)\in N(i,j,d)} \left(B_{ij}, \frac{\sigma_w^2}{\|Q(\alpha,\beta,0)\|} + q_{i,j,d} \right) \cdot \|Q(\alpha,\beta,0)\|}{\sum_{(\alpha,\beta)\in N(i,j,D)} \|Q(\alpha,\beta,0)\|} \tag{7.27}$$

We identify the desired variance σ_N^2 as the variance of $\hat{B}_{ij}$; that is, the variance of (7.27) is the sought estimation error variance

$$\sigma^2_{N(i,j,D)} = \frac{\sum_{d\in D} \left(\sum_{(\alpha,\beta)\in N(i,j,d)} \sigma_w^2 \cdot \|Q(\alpha,\beta,0)\| + q_{i,j,d} \cdot \|Q(\alpha,\beta,0)\|^2 \right)}{(\sum_{(\alpha,\beta)\in N(i,j,D)} \|Q(\alpha,\beta,0)\|)^2} \tag{7.28}$$

This cumulative variance $\sigma^2_{N(i,j,D)}$ can be computed for each of the selected neighbourhoods $D_0, \ldots, D_T$, so the neighbourhood for each spatial location i,j is selected by minimizing $\sigma^2_{N(i,j,D_t)}$ over index t.

7.5 Results

We have applied the proposed background estimation method to different sequences of phase contrast HSC images. The first and 50th frames of one of HSC image sequences were depicted in Fig. 7.1. The minimum cumulative variance computed for each spatial location of this image sequence is depicted in Fig. 7.11(a). The selected window size for each spatial location based on computed cumulative variance (Fig. 7.11(a)) is depicted in Fig. 7.11(b). The estimated background applying the proposed method to this sequence is depicted in Fig. 7.11(c). As we can observe in the estimated background, not only is the well boundary precisely estimated, but there are very smooth variations over the background image. The background subtracted images for frames 1 and 50 are depicted in Fig. 7.12. Cell boundaries are accurately preserved while we can observe a uniform background everywhere else.

The proposed mutual method is applied to imperfectly cropped image sequences in which well boundaries are visible. The estimated background applying the first pass of the proposed background estimation method (point-wise), corrected image frame after background subtraction, and the located cell centers applying the proposed cell model (6.23) in Sec. 6.4 are depicted in Figs. 7.13. The remaining number of samples for each spatial location, the temporal variance of each spatial location, and the computed minimum cumulative variance for each spatial location are depicted in Figs. 7.14. The window size selection based on the computed cumulative variance, the precise estimated background applying the second pass of the proposed background estimation method (window-wise), and the background subtracted image frame are depicted in Figs. 7.15. As we can observe in Figs. 7.15, the proposed method effectively has removed the stationary objects and has retained the important foreground elements such as boundaries of quasi-stationary cells. Cell centers are located (Fig. 7.13(c)), and a precise smooth background image is estimated

(Fig. 7.15(c)). Fig. 7.16 shows the estimated background and the corrected image frames for different HSC phenotypes for which the first pass of background estimation/subtraction (point-wise) were depicted in Fig. 6.10.

The results obtained by the proposed method is compared with the other methods as follows. First, we implemented a non-adaptive non-segmentation method in [35] which is a modified version of [126] which we call it *V2*. In comparison with our proposed method, *V2* is also spatio-temporal, however in contrast with our proposed method which is adaptive multi-clique, *V2* is pointwise and does not use the segmentation information to remove the foreground objects. The estimated background images applying $V2$ and the proposed method are depicted in the top row, Figs. 7.17(Left) and 7.17(Right) respectively. As can be observed, wherever the foreground objects have slow motion dynamics, $V2$ fails to precisely estimate the background, and cell boundary pixels are visible in the estimated background.

Second, the proposed method by *Heikkila and Pietikainenin* [63], a recent background modelling method with very promising results is implemented. To train the background model using LBP (Local Binary Pattern) texture operator we use the same number of $K = 100$ frames as we used to estimate the background in the proposed method and $V2$. To have more accurate results, small values for training parameters are selected as $\alpha_b = 0.05$ and $\alpha_w = 0.05$. $LBP_{R,P}$ with $R = 8$ & $P = 2$ is used as texture operator to generate the results. After the background model is trained, the same sequence of images $\mathbf{I} = (I_1, I_2, \cdots, I_K)$ that is used in the training step is classified. In the test step each pixel of each frame is classified to foreground or background and finally the background image is estimated over the K frames considering pixels which have been classified as background pixels. We then compared the estimated background images applying *Heikkila and Pietikainenin* and the proposed method as are shown in the top row, Figs. 7.17(Middle)

and 7.17(Right) respectively. As can be observed, where cells have slow dynamics *Heikkila and Pietikainenin* also fails to accurately estimate the background, in contrast with the proposed method that precisely estimates background and generates very smooth, uniform background. Cell boundary pixels are apparent in the estimated background images using LBP based method and $V2$.

We then compared the background corrected sequences obtained by applying three mentioned methods. To do so, the original image sequence is corrected subtracting the background image estimated by each method. The results for $V2$, *Heikkila and Pietikainenin* and the proposed method are depicted in Fig. 7.17. In estimated background images depicted in Figs. 7.17(Left) and 7.17(Middle), cell boundary of quasi-stationary cells are visible in the estimated background, so as we expect, the bright boundary of quasi-stationary cells are significantly degraded in the resultant images after background subtraction which are apparent in Figs. 7.17(Left) and 7.17(Middle). However the cell boundaries are preserved applying our proposed method as it is obviously visible in Fig. 7.17(Right).

As the third and the forth approaches, a frame-difference segmentation method and a morphological averaging background estimation method in [75] are developed. Depicted in Figs. 7.18 and 7.19, we can observe that neither frame-difference segmentation nor morphological averaging background estimation provide satisfactory result. Segmented cells are depicted as some scattered pixels in Fig. 7.18 using the frame-difference method. As we can observe, segmented foreground pixels do not maintain any recognizable shape. The main disadvantage of the segmentation methods based on frame-difference is that they cannot discriminate quasi-stationary objects from the background, thus slow moving objects will be eliminated by subtracting the consequent frames.

The segmented foreground that is obtained by applying the method in [75] is more concentrated in cell areas in comparison with the frame-difference method and it maintains

some scattered cell boundaries, yet it cannot locate separated cells as can be observed in Fig. 7.19.

In contrast with these methods, applying the proposed cell model (6.23) in Sec. 6.4 can perfectly locate the cell centers in each frame. The result is depicted in Fig. 7.13(c) for the same frame as we applied the frame-difference and morphological background estimation methods.

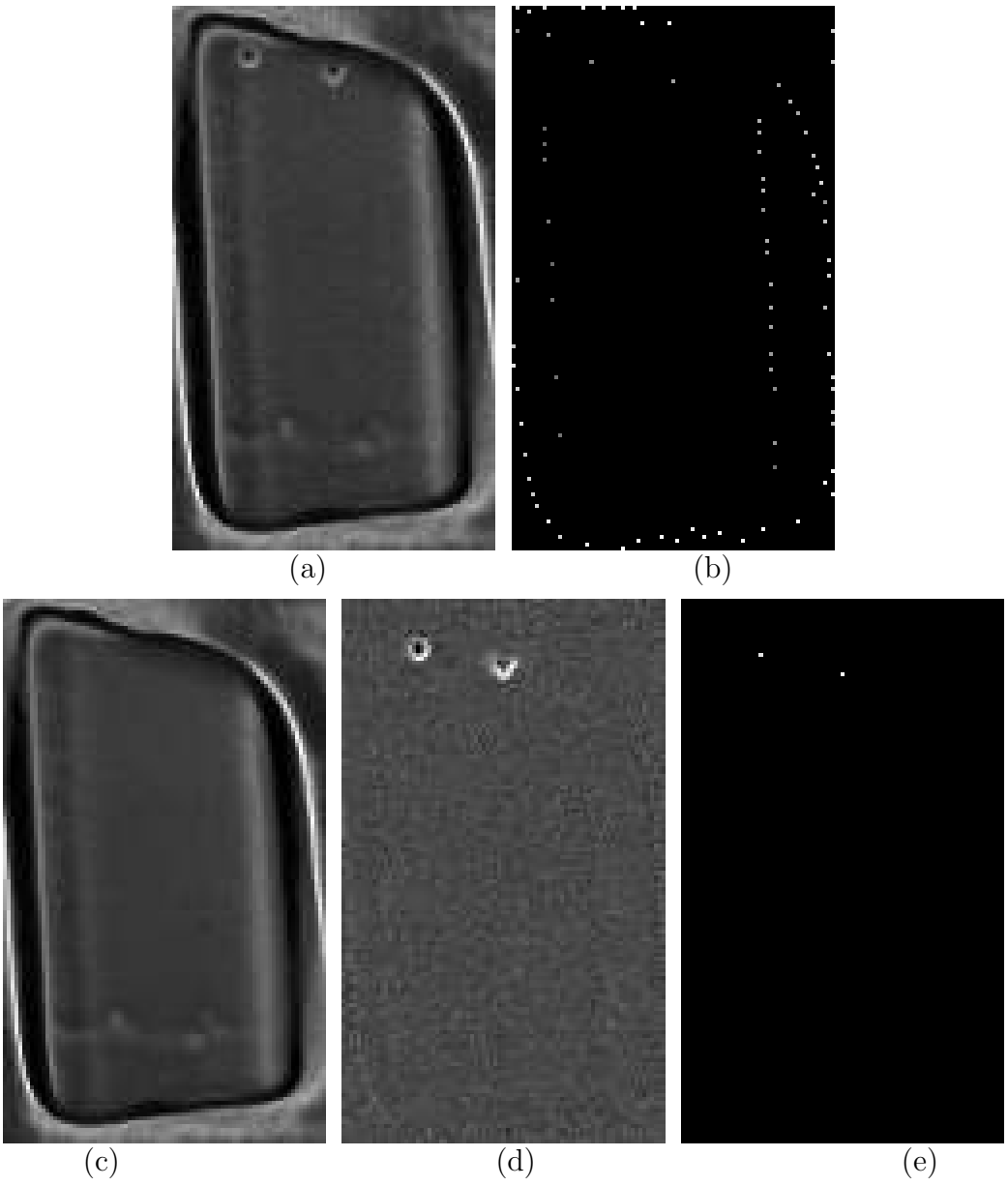


Figure 7.4: (a) Coarsely cropped well in which well boundaries are visible. (b) Applying the cell model in (6.2), to the coarsely cropped well. In contrast with the perfect result obtained by applying cell model (6.2) to a cropped well interior without boundaries (Fig. 7.2(b)), here the cell model (6.2) performs very poorly where the cropped well contains visible well boundaries. (c) The estimated background obtained by applying the pointwise method presented in Sec. 7.3.1. (d) Panel (a) after background subtraction. (e) Applying the cell model (6.23) to (d).

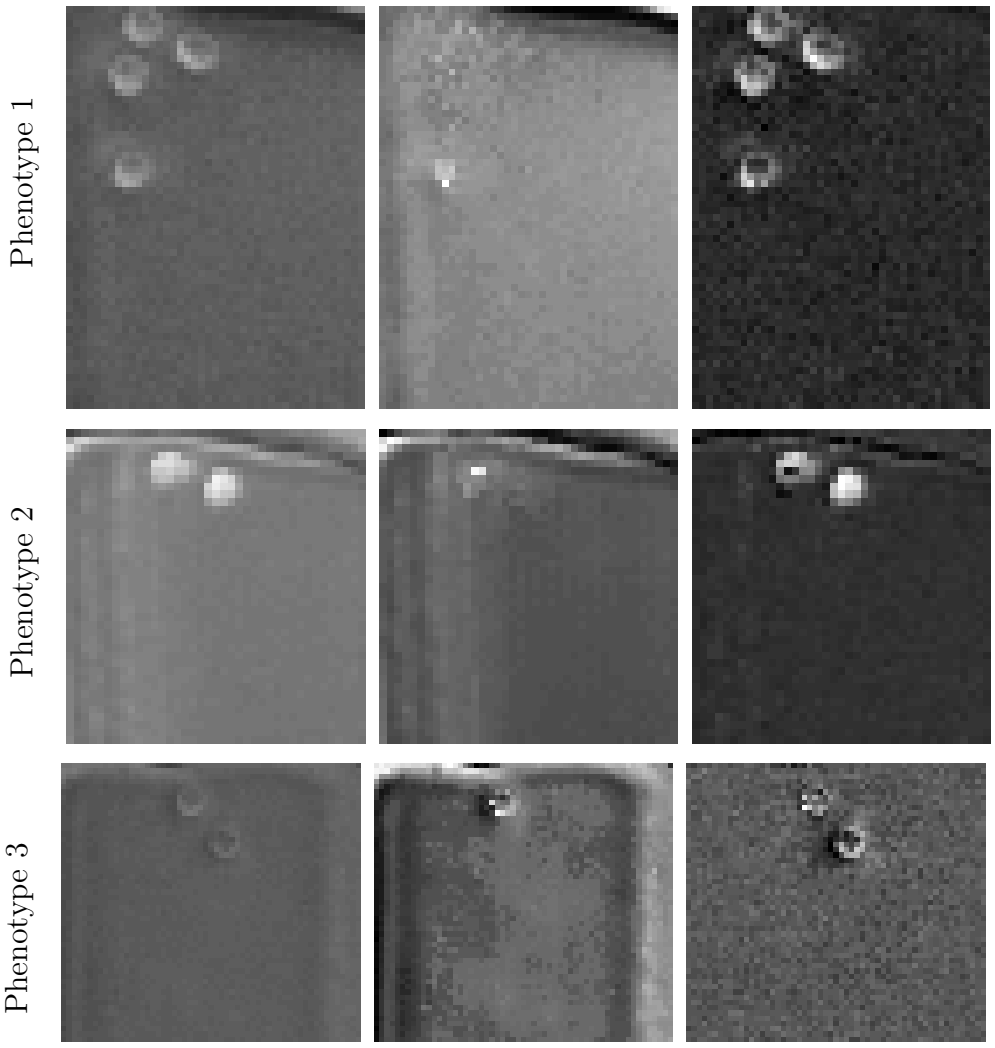
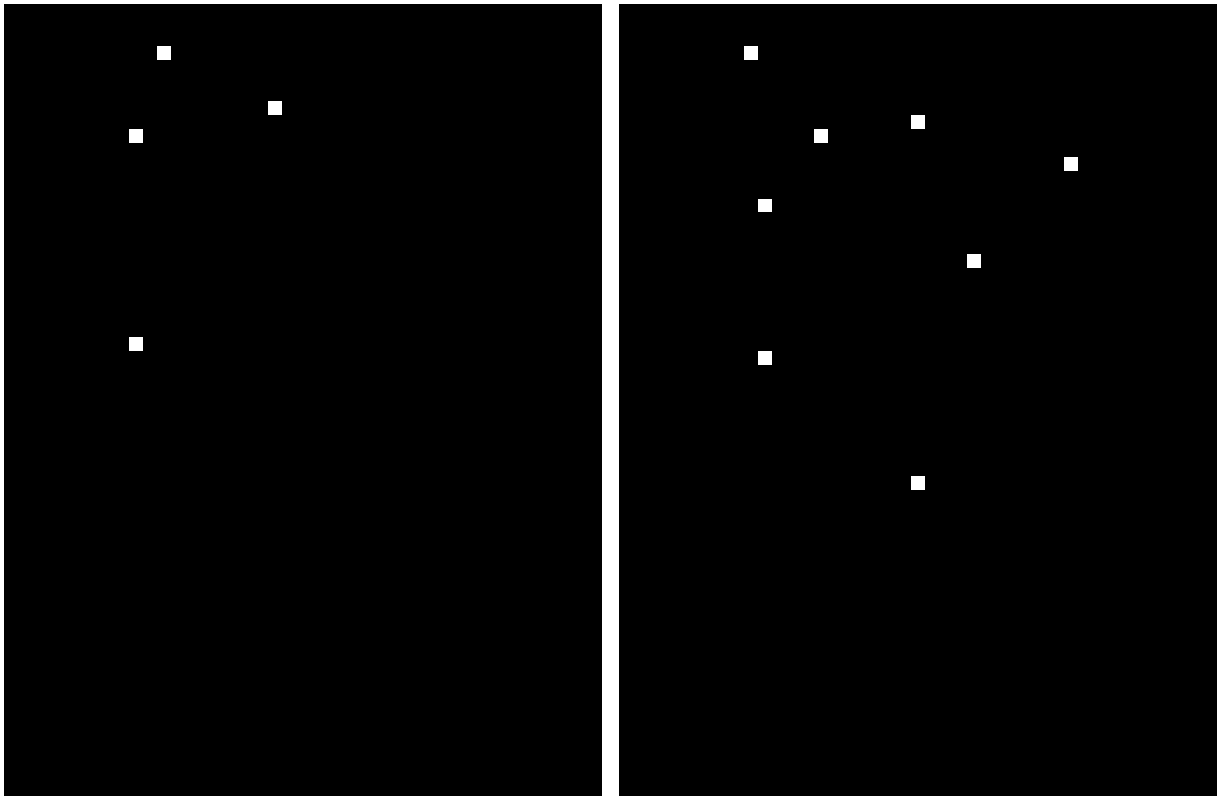
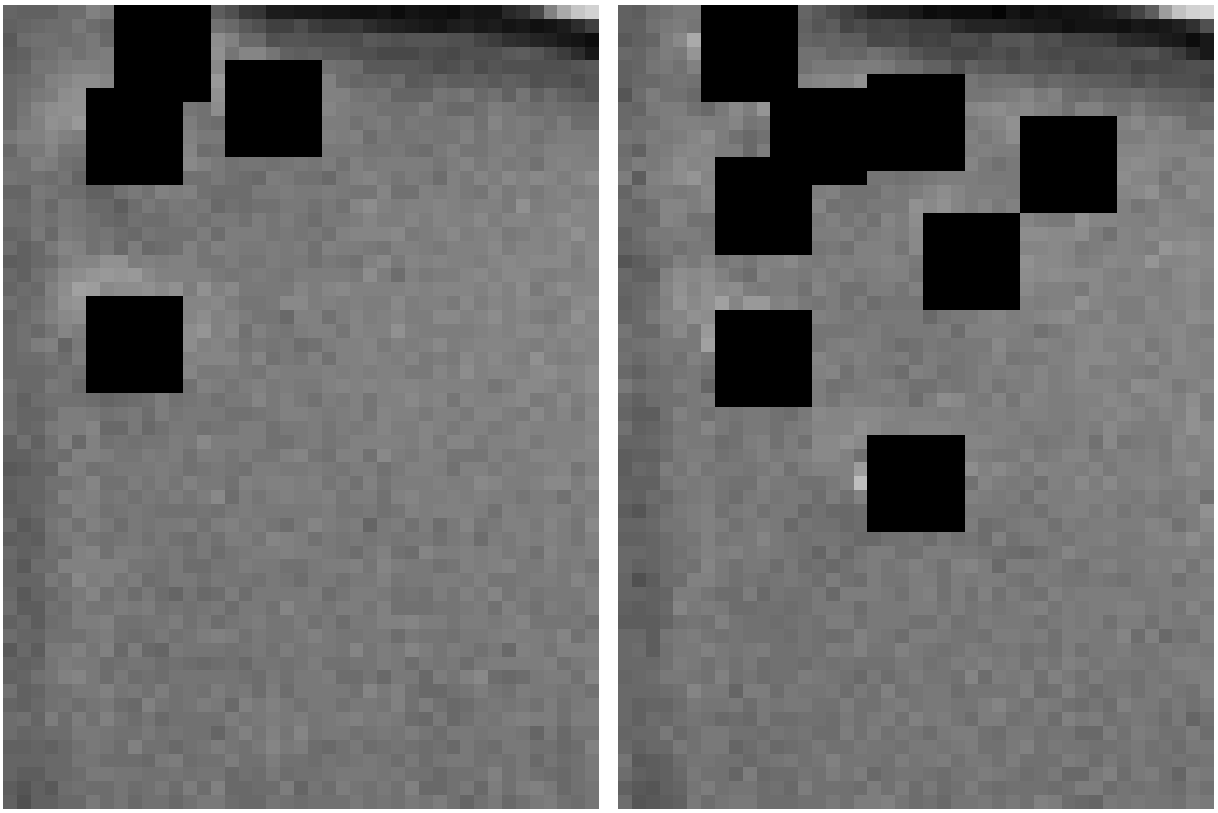


Figure 7.5: Estimated background images depicted in the second column for different HSC phenotypes are obtained by applying the pointwise method presented in Sec. 7.3.1. Well-boundaries and the stationary background pixels are well estimated applying the pointwise background estimation method (Sec. 7.3.1). However quasi-stationary cells as can be observed in all three background images, leading to cell-background contrast loss in the background-subtracted images.



(Frame 1) (Frame 50)

Figure 7.6: Cell center locations obtained by applying the cell model in Sec. 6.4 to the background subtracted images for which the original images are depicted in Fig. 7.1.



(Frame 1) (Frame 50)

Figure 7.7: The original images after removing the foreground cells detected in Fig. 7.6. Observe how the imperfect segmentation of Fig. 7.6 causes slight cell removal failure in the top left and bottom cells in the Frame 50.

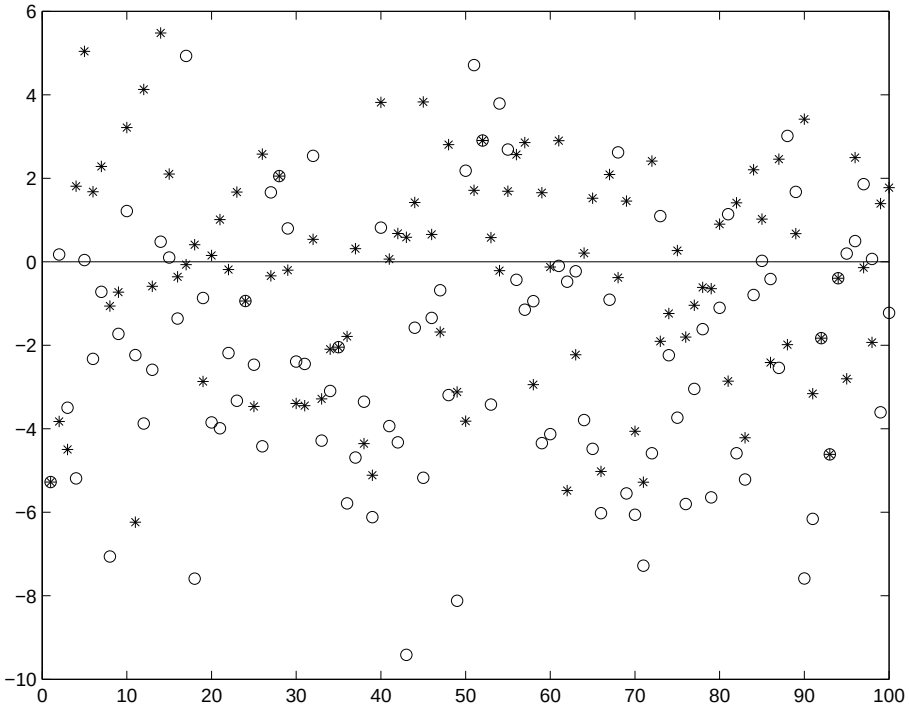


Figure 7.8: Because of imperfect segmentation, cell fragment pixels appear as outliers in the background samples after removing the segmented cells from the image frame. A cell fragment pixel over 100 frames is depicted as circles (o) while intensity variations and *mode* of a typical background pixel are depicted as stars (*) and solid line respectively.

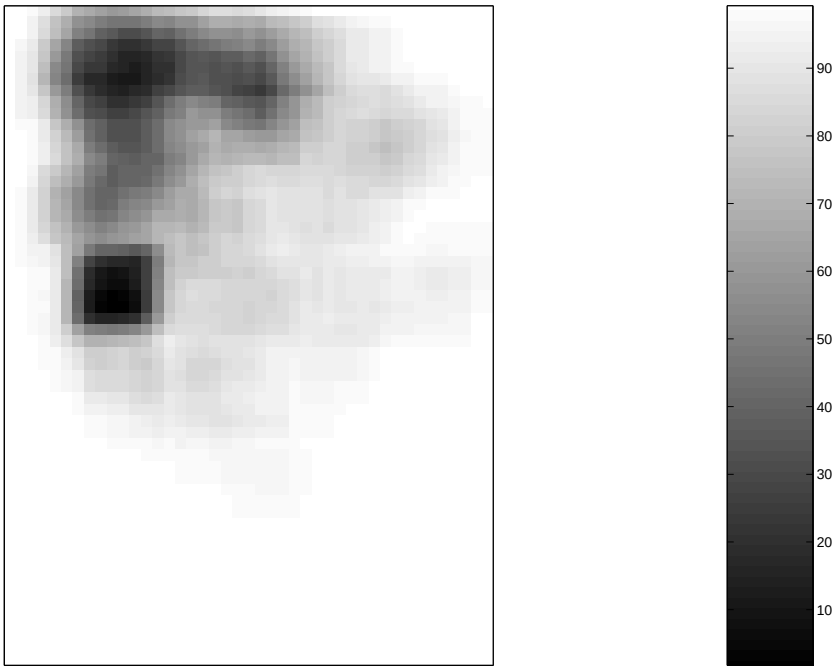
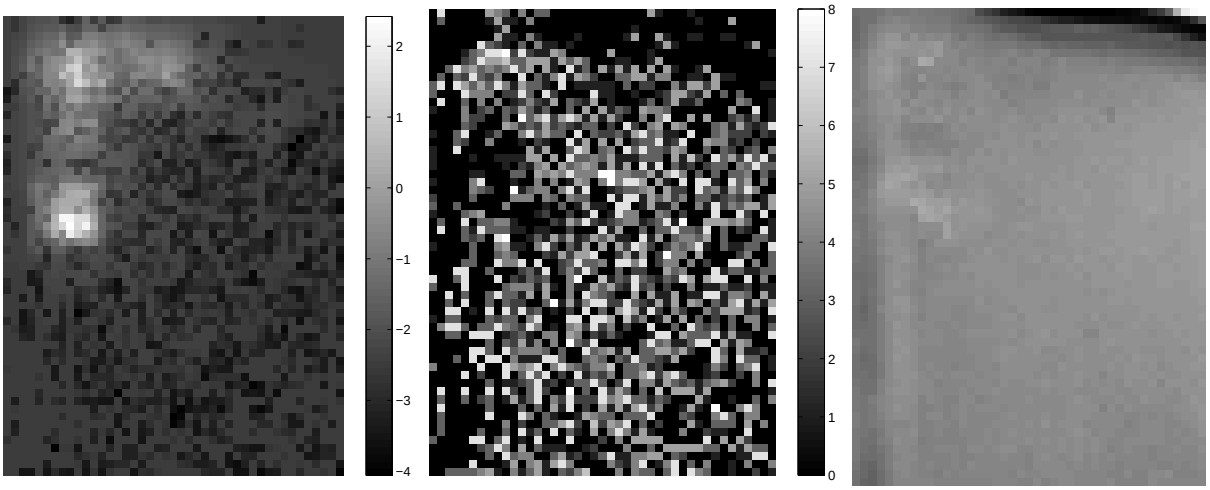


Figure 7.9: A visual representation of the number of samples available, shown by gray level, for background calculation. The presence of cells in the top half of the image (Fig. 7.1) has the effect of reducing the number of background samples there. As some pixels in the upper left have few background samples remaining, the background needs to be computed over some spatial extent of pixels in a neighbourhood to increase the number of samples.

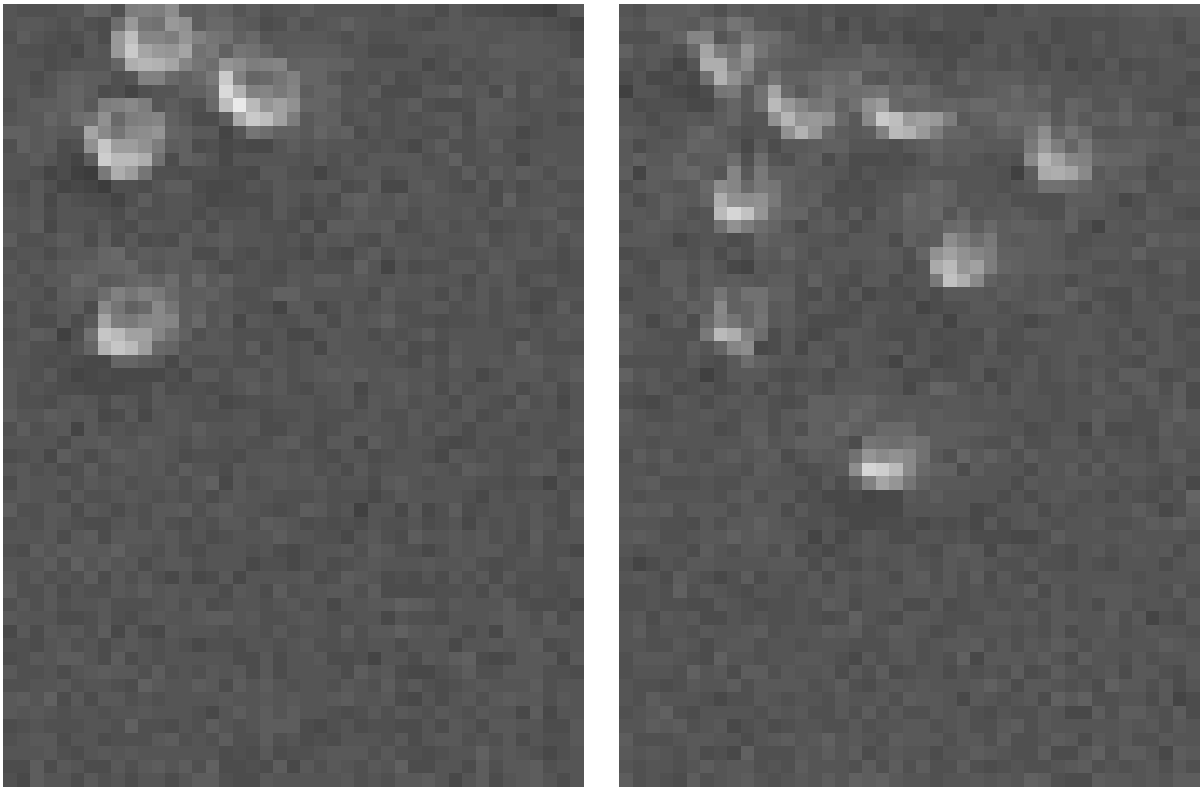
		9	9	9		
	8	8	7	6	6	
9	8	4	3	2	6	9
9	5	1	0	1	5	9
9	6	2	3	4	8	9
	6	6	7	8	8	
		9	9	9		

Figure 7.10: Clique sets $c(\Delta i, \Delta j) \in \{0, 1, \cdots, 9\}$ where Δi and Δj are spatial offsets with respect to the central location.



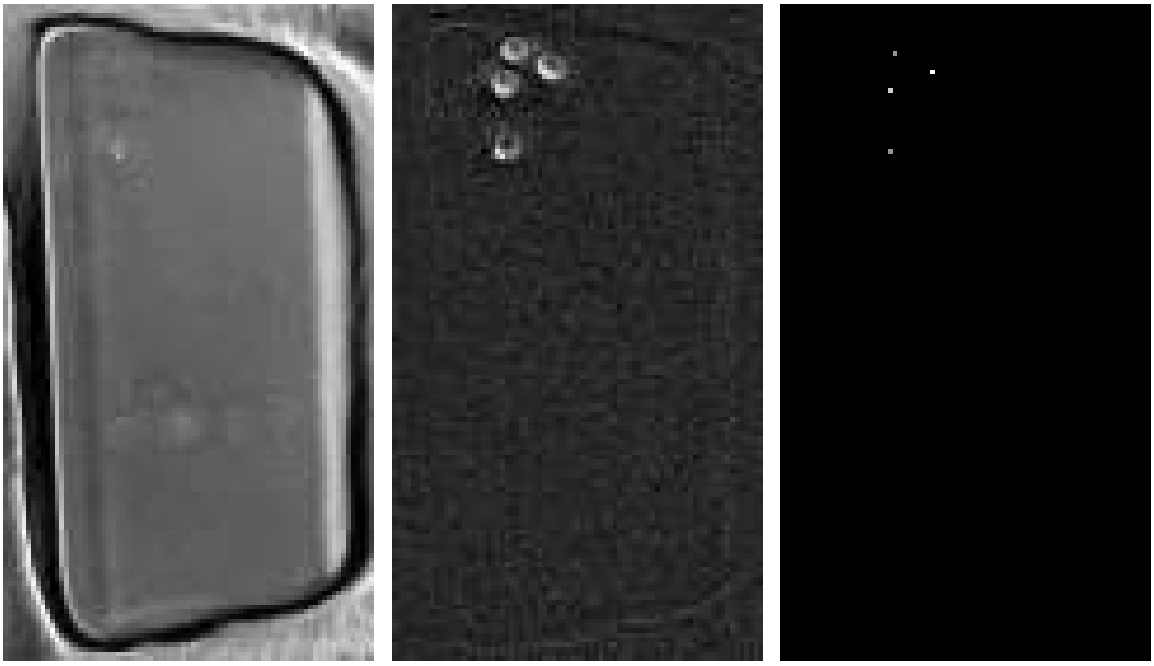
(a) (b) (c)

Figure 7.11: The proposed multi-clique background estimation method presented in Sec. 7.3.2: (a) Logarithm of minimum cumulative variance $log(\sigma^2_{N(i,j,D)})$ computed for each spatial location of image sequence depicted in Fig. 7.5 for Phenotype 1. (b) Selection of window size based on computed cumulative variance. (c) Estimated background.



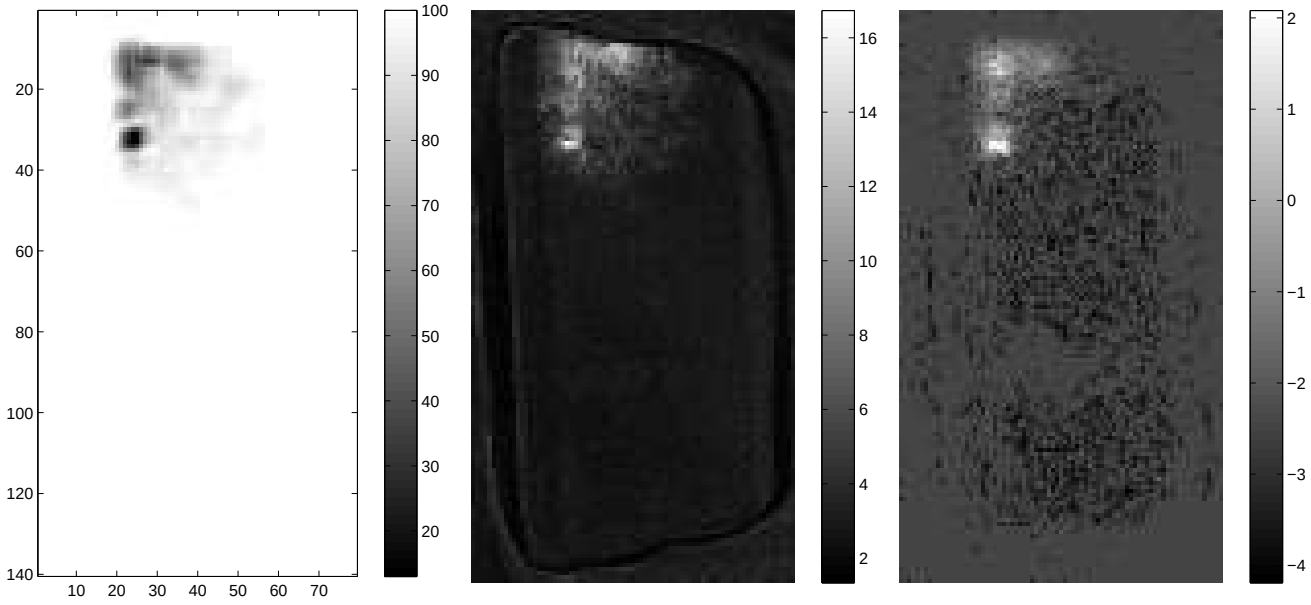
(Frame 1) (Frame 50)

Figure 7.12: Subtracting out the estimated background image in Fig. 7.11(c) applying multi-clique method from the original image in Fig. 7.1.



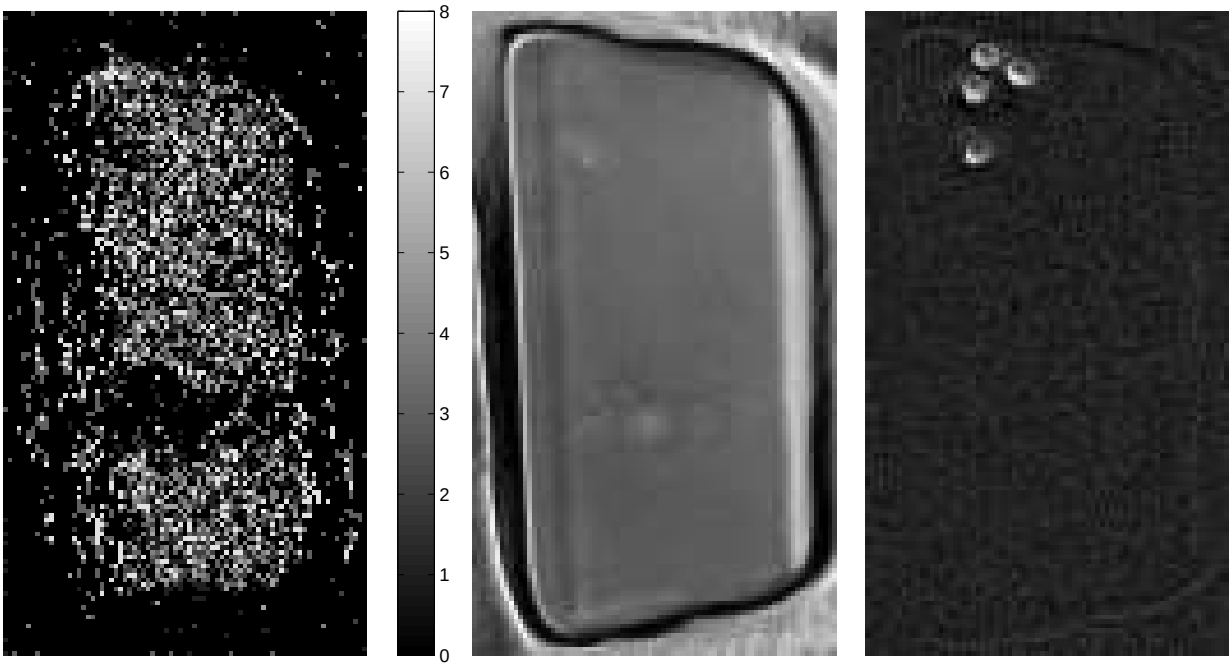
(a) (b) (c)

Figure 7.13: (a) Pointwise estimated background applying the method discussed in Sec. 7.3.1 for image sequence with typical frame depicted in Fig. 7.18. (b) Subtracting out the estimated background image obtained in (a) from the original image. (c) Located cell centers applying cell model P in (6.23) from Sec. 6.4.



(a) (b) (c)

Figure 7.14: (a) A visual representation of the number of samples available shown by gray level. (b) Temporal variance of each spatial location. (c) Logarithm of minimum cumulative variance $log(\sigma^2_{N(i,j,D)})$ computed for each spatial location.



(a) (b) (c)

Figure 7.15: (a) Selection of neighbourhood based on computed cumulative variance. (b) Estimated background applying the multi-clique method proposed in Sec. 7.3.2. (c) The final, corrected image subtracting out the estimated background.

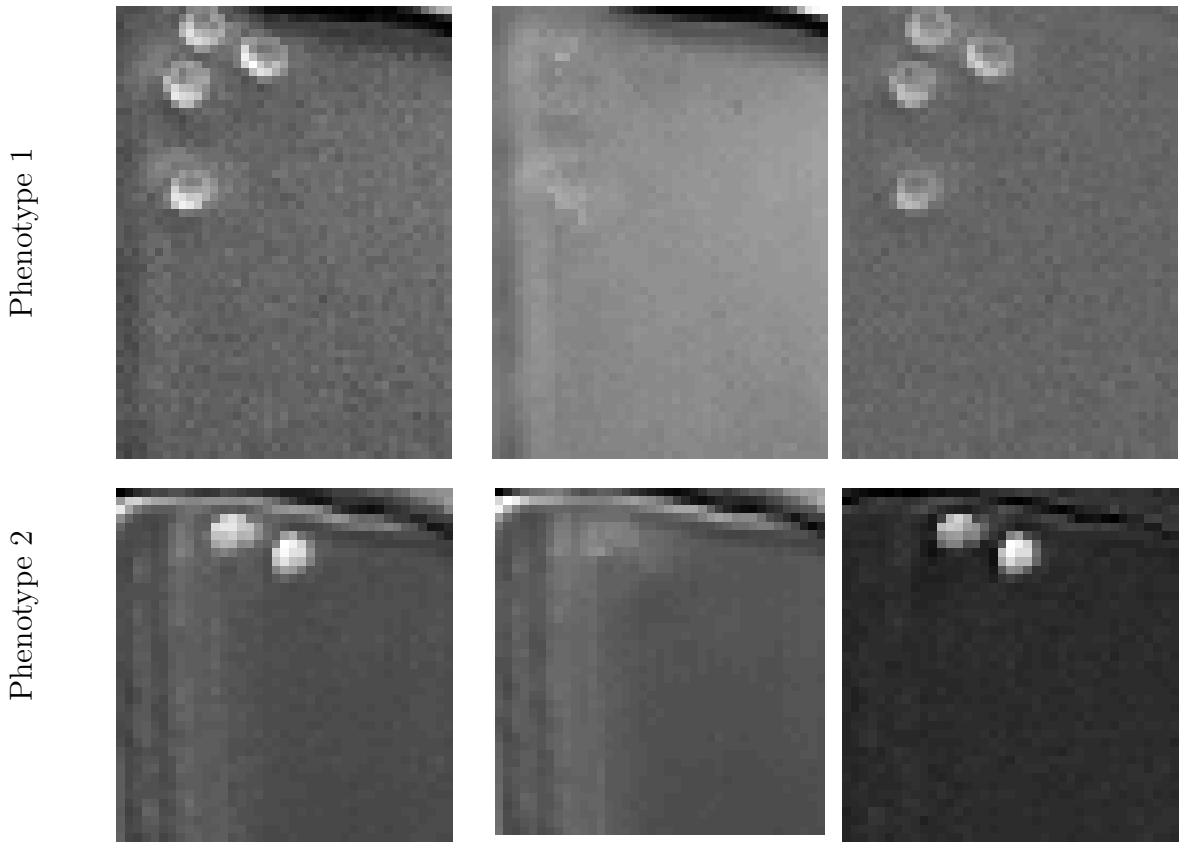


Figure 7.16: Estimated background images by applying the multi-clique method of Sec. 7.3.2. Not only are well-boundaries and stationary background pixels well estimated, but quasi-stationary cells present in all three images do not appear in the estimated background. Rather, the quasi-stationary cells are well separated from the background by removing the segmented cells and estimating the background over a multi-clique neighbourhood. Therefore the corrected images obtained by subtracting the estimated background images do not suffer and cell-background contrast is maintained or enhanced as we can see in the corrected images in the third column.

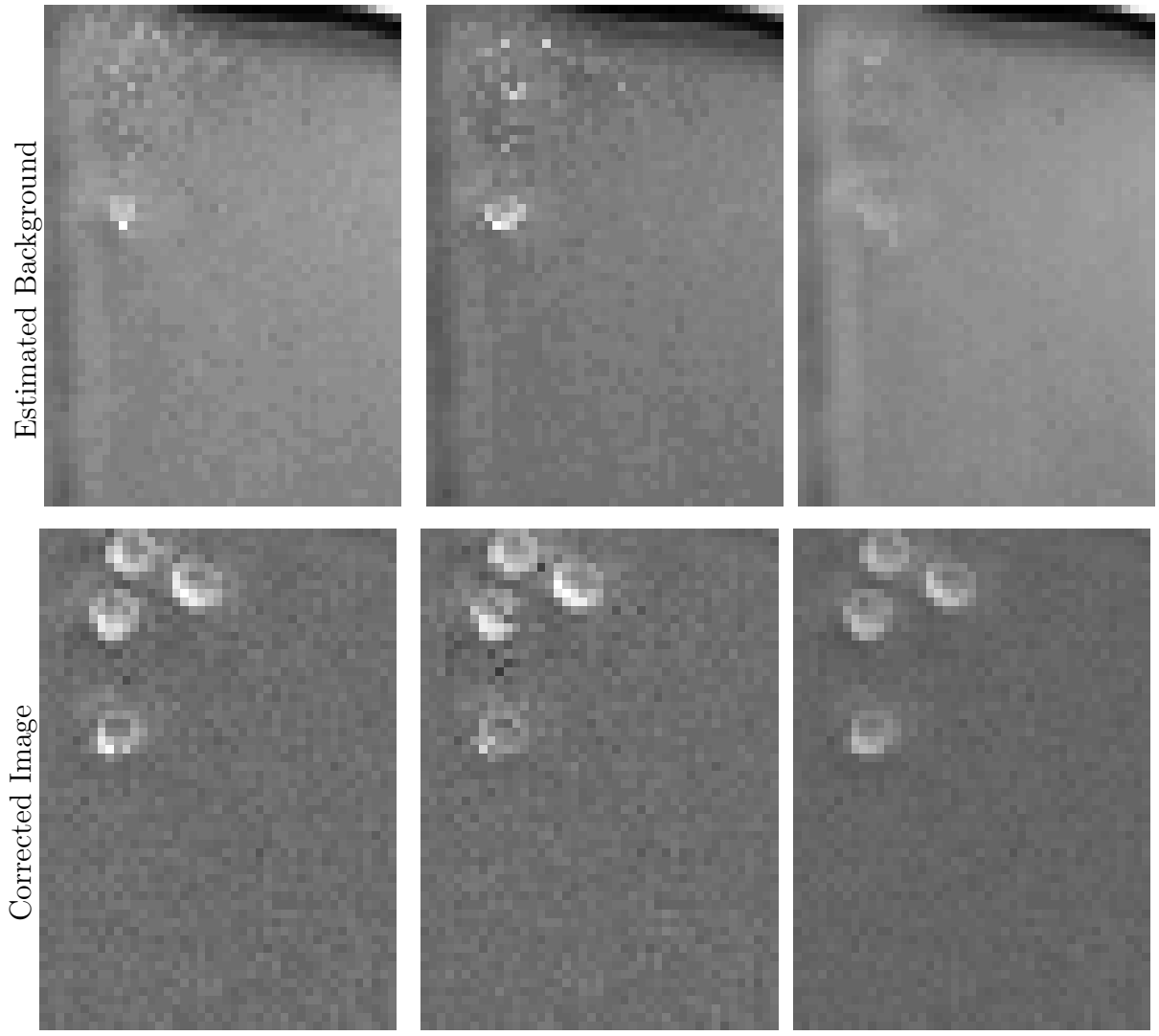
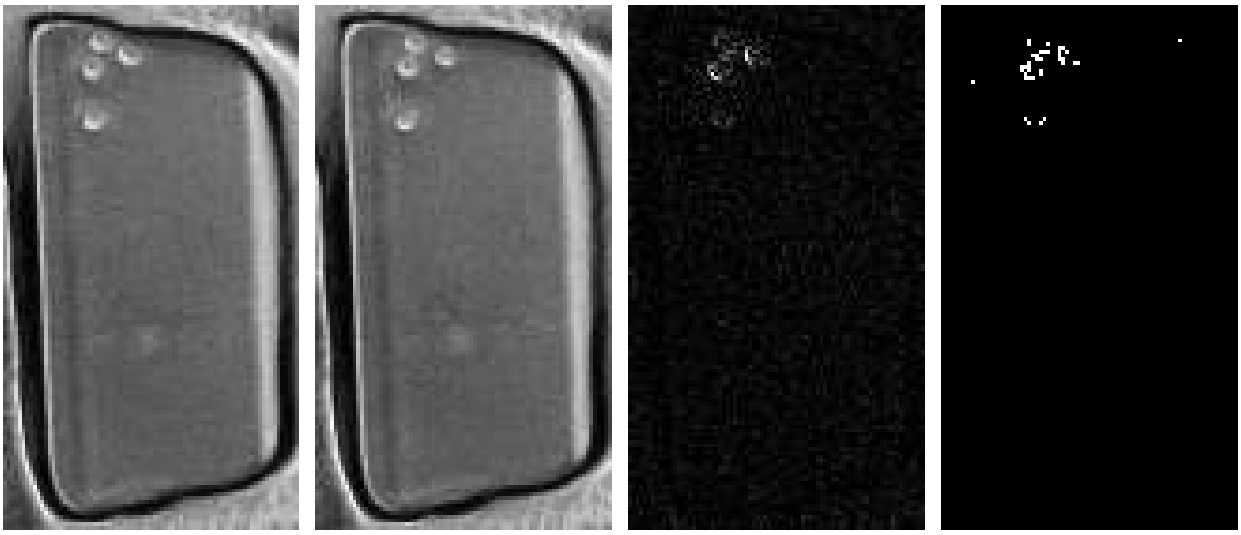


Figure 7.17: Estimated background and corrected images are depicted for Phenotype 1, based on applying *V2*, [63], and the multi-clique method proposed in Sec. 7.3.2. The presence of quasi-stationary cells in the mid-left of the image sequence causes *V2* and [63] to interpret the cells as belonging to the background, leading to a cell-background contrast loss in such locations after background subtraction, as we can see in the corrected images. Our proposed multi-clique method, shows in the rightmost column, has quasi-stationary cells only barely appearing in the estimated background.



(a) (b) (c) (d)

Figure 7.18: (a) A typical frame k. (b) Frame $k+1$. (c) Difference of frames k and $k+1$. (d) Segmentation of foreground cells based on consequent frames differences. It can be observed that frame-difference segmentation does not yield a satisfactory segmentation result. Segmented cells are depicted as some scattered pixels using frame-difference method where segmented foreground pixels do not maintain any recognizable shape.

(a) (b) (c) (d)

Figure 7.19: (a) A typical frame k. (b) Estimated background using morphological operators. (c) Subtracted background from frame k. (d) Segmented frame k. The segmented foreground that is obtained by applying the morphological background estimation/subtraction method is more concentrated in cell areas in comparison with the frame-difference method and it maintains some scattered cell boundaries, yet it cannot locate separated cells as can be observed.

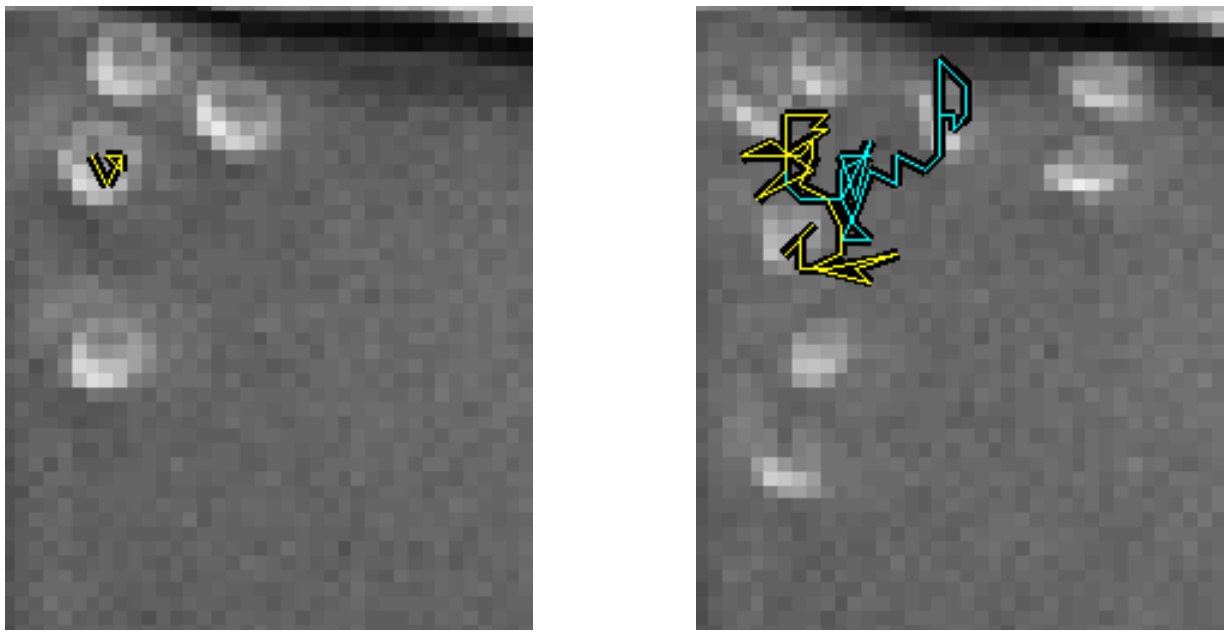
Chapter 8

Cell Tracking

Multi-target tracking, the association of detected points into sequences over time, is an important NP-hard problem. Considerable efforts have been conducted to design tractable methods by reducing its complexity. These methods as were discussed in Sec. 3.4 include Nearest Neighbor [31], Joint Probabilistic Data Association [7, 114] and Multi-Hypotheses Data Association [105, 114]. The common task among all tracking methods is to reduce the hypothesis space, the set of plausible association solutions, and to solve the association problem by selecting the most likely hypothesis, normally yielding a suboptimal solution. Solving the association problem in a reduced hypothesis space raises some important questions, such as the likelihood of finding the optimal solution in the reduced space, and the closeness of the optimal solution to the reduced space.

We are interested in the tracking of Hematopoietic Stem Cells (HSCs) in culture in this research. A small fraction of a typical HSC microscopic image is depicted in Fig. 8.1 with the superimposed dynamics of a mature blood stem cell before and after splitting.

In this chapter the third important part of our cell tracking system, target-measurement association, will be introduced and discussed.



(a) (b)

Figure 8.1: Close-up of a HSC phase contrast microscopic image with the superimposed track of one mature blood stem cell (a) 8 frames before and (b) 30 frames after splitting.

8.1 MAP Estimation

Aside from the cell radius, the blood stem cells which are of interest here all have the same visual appearance and cannot be discriminated visually. Therefore to track a particular cell over time, the association task becomes crucial.

With a model in place in Sec. 6.4 describing the spatial pattern of pixels with the appearance of a cell, we move to the core of the problem: given a sequence of images $I_{1:K} = \{I_1, I_2, ..., I_K\}$ and a definition of our "target" (cell models in chapter 6), we need to associate the cells over time. Denote by $F_{1:K}$ a possible hypothesis of the K-frame association problem,

$$F_{1:K} = \{f_1, f_2, ..., f_K\} \tag{8.1}$$

where f_k is a parametric representation of frame k. In the case of HSCs, f_k is defined as

$$f_k = \{(l_k^j,\ u_k^j,\ r_k^j,\ s_{k,j}),\ 1 \leq j \leq M_k\} \tag{8.2}$$

where j indexes the M_k cells present in frame k, l_k^j is the label of the associated parent cell in frame $k-1$, $u_k^j = (x_k^j, y_k^j, r_k^j)$ specifies the cell radius r_k^j and location (x_k^j, y_k^j), and s_k^j is the cell age, updated as

$$s_k^j = \begin{cases} 1 & \text{if } \exists i \text{ such that } l_k^j = l_k^i \ \text{ (i.e. cell split)} \\ (s_{(k-1)}^{l_k^j}) + 1 & \text{Otherwise.} \end{cases} \tag{8.3}$$

The cell dynamics, affecting the relationship of (x_k^j, y_k^j) with $(x_{k-1}^{l_k^j}, y_{k-1}^{l_k^j})$ will, in general, be cell-type specific and may further be influenced by environmental factors, chemical gradients etc. In our context there are no deliberate experimental biases, and successive image

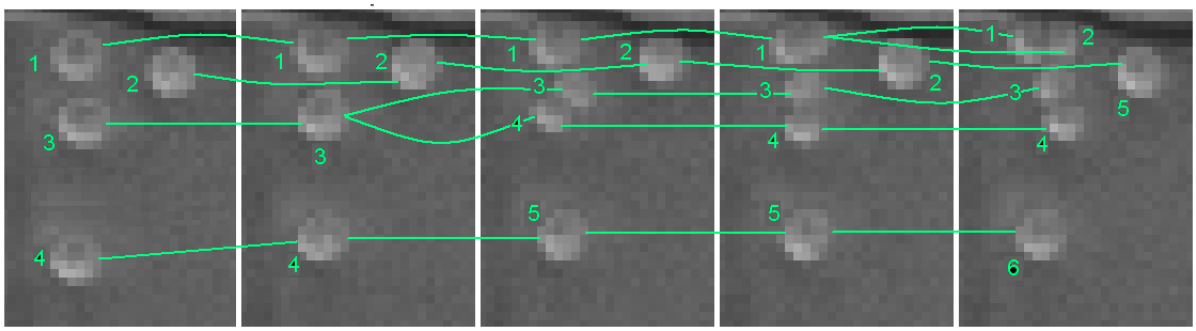


Figure 8.2: An illustration of cell association over time with numeric labels.

frames are spaced so far apart in time (several minutes), therefore inter-frame cell motion is independent over time, and a Gaussian random walk was found to well-approximate hand-tracked cell motion.

The goal is to solve the spatio-temporal cell segmentation-association problem of Fig. 8.2: we wish to estimate $F_{1:K}$ given the image sequence $I_{1:K}$ and given an initialization f_0 in frame zero.

8.1.1 Solution Of NP-Hard Problems

The optimum answer to the association problem is the maximum a posteriori estimation of $F_{1:K}$

$$\hat{F}_{1:K} = arg \left\{ \underset{F_{1:K}}{max} \; P(F_{1:K} |\; Z_{1:K}) \right\} \tag{8.4}$$

which is NP-hard, with a complexity exponential in K, so to find the optimal solution is essentially impossible. Instead, to solve the association problem, different methods have been introduced to solve the problem sub-optimally either by finding the most likely hypothesis from a limited hypothesis set over multiple frames

$$\{F^h_{1:K} |\; h = 1, 2, ...\} \tag{8.5}$$

thus finding the best member

$$\hat{F}_{1:K} = F^{\hat{h}}_{1:K} \quad \text{where} \quad \hat{h} = arg \left\{ \underset{h}{max} \;\; P(F^{h}_{1:K}|Z_{1:K}) \right\} \tag{8.6}$$

as the solution such as the Multi Hypothesis Tracking (MHT) algorithm (Sec. 3.4.5) [105], or a frame by frame solution based on single-frame associations over time $[k-1, k]$ such as Joint Probabilistic Data Association (JPDA) (Sec. 3.4.3) [114].

The single-frame association method is a feasible approach which has been widely used. However even in the single-frame case, virtually all approaches propose to find solutions over a reduced hypothesis space for frame k:

$$\{f^{h}_{k} |\; h = 1, 2, ...\} \tag{8.7}$$

and searching for the best member

$$\hat{f}_{k} = f^{\hat{h}}_{k} \quad \text{where} \quad \hat{h} = arg \left\{ \underset{h}{max} \;\; P(f^{h}_{k}|Z_{1:k}) \right\} \tag{8.8}$$

as the solution. The optimal single-frame solution is found if it is included among the hypotheses of that frame, i.e., if

$$arg \left\{ \underset{f_k}{max} \;\; P(f_k|\; Z_k) \right\} \in \{f^{h}_{k}\} \tag{8.9}$$

The key, here, to efficiency is to minimize the number of hypotheses; the key to the quality of estimation is to include many likely hypotheses in the hypothesis set. As these goals are in opposition, we are left with a complexity/quality trade off. One of the widely used and well known single-frame algorithms to solve the multi-target tracking problem in a reduced hypothesis space is JPDA [114].

8.1.2 The Proposed Method: An Optimal Single Frame Tracking

As was mentioned in chapter 3, JPDA has been extensively used as a single-frame solution for multi-target tracking problem. Most of the single-frame tracking methods, includ-

ing JPDA, reduce the number of possible association hypotheses to make the association problem tractable. The major shortcoming of these methods is finding the sub-optimal single-frame solution over a reduced hypothesis space. The other disadvantages of JPDA are the assumptions of linear dynamics and Gaussian measurements.

An optimal general-purpose single-frame approach must solve the association problem over the entire single-frame hypothesis space and assume no restrictions on dynamics and measurements. To achieve this goal and to derive an algorithm to be applicable to the stem-cell problem at hand in which dynamics and measurements are respectively assumed non-linear and non-Gaussian, a non-linear, non-Gaussian method is proposed to solve the tracking problem. A feasible optimal method is proposed to evaluate all possible hypotheses, by representing the tracking problem in the form of an assignment problem and then by extending a linear programming optimization method known as the Hungarian method [76] to apply to the problem at hand. Network programming and optimization, as a branch of operations research, has seen considerable research attraction. A special case of this class of problems is the assignment problem, which has been solved using a class of linear programming methods known as primal-dual algorithms [95], of which the Hungarian method is an example. An assignment problem of order n is the one-to-one connection or assignment of n sources to n sinks. The solution can be represented by a binary matrix of order $n \times n$, as illustrated in Fig. 8.3 for $n = 3$.

In this section we will represent the tracking problem in the form of an assignment matrix by considering the cells (previous frame) and measurements (current frame) as the row and column indices respectively. First the proposed non-linear, non-Gaussian method follows, then we discuss how the proposed formula can be represented and solved as a generalized assignment problem.

		Sinks.		
		A	B	C
Sources	1	1	0	0
	2	0	0	1
	3	0	1	0

Figure 8.3: Example Assignment Matrix: Three sources $1, 2, 3$ need to be assigned to three sinks A, B, C. Clearly each row and column of the matrix must sum to one.

8.1.3 Non-Linear, Non-Gaussian JPDA (NNJPDA)

In standard JPDA new tracks can not be initiated, the number of targets is known, dynamics are linear, measurements are Gaussian and the Kalman filter is used to predict the new state. In contrast, in blood stem cell tracking new tracks for divided cells must be initiated, the number of cells is unknown, the dynamics are a constrained random walk (non-linear due to motion constraints imposed by nearby cells), and due to clutter the measurements are non-Gaussian. The non-linear, non-Gaussian nature of problem makes the Kalman filter an inappropriate choice to predict the new state.

The basic measurement constraints are as follows: i) Each measurement originates from only target (cell) or clutter; ii) Each measurement can be associated to one cell; iii) Up to two measurements in frame k can be associated to the same cell in frame $k-1$, where this last constraint differs from standard JPDA.

Each cell in the state f_k must belong to one of the following sets:

Unassociated: $U = \{j|\ l_k^j = 0\}$

Split: $S = \{j|\ l_k^j = l_k^i \text{ for } j \neq i\}$

Regular: $R = \{j|\ j \notin \{U \cup S\}\}$

To evaluate the association solution $\hat{f}_k$, a measurement set Z_k is obtained to update the previous state estimate $\hat{f}_{k-1}$ by selecting the hypothesis with the maximum joint association probability among all possible hypotheses for that frame.

Recall the recursive Bayesian estimator (3.33) from chapter 3

$$P(f_k|\ Z_{1:k}) = \lambda_k \cdot P(f_k|\ Z_{1:k-1}) \cdot P(Z_k|\ f_k) \tag{8.10}$$

conditioning with f_{k-1} on both sides we have

$$\begin{aligned} &P(f_k|\ Z_{1:k}, f_{k-1}) = \\ &\lambda_k \cdot P(f_k|\ Z_{1:k-1}, f_{k-1}) \cdot P(Z_k|\ f_k, f_{k-1}) = \\ &\lambda_k \cdot P(f_k|f_{k-1}) \cdot P(Z_k|\ f_k) \end{aligned} \tag{8.11}$$

where the first term explicitly examines the likelihood of f_k given f_{k-1}, which involves a dependence on cell dynamics, splitting, and separation probabilities:

$$\begin{aligned} &P(f_k|\ f_{k-1}) = \\ &\lambda_a \cdot (Dynamics) \cdot (Splitting) \cdot (Separation) \end{aligned} \tag{8.12}$$

The second term of (8.11) assesses the likelihood of the measurement Z_k given the cell state f_k. Because, in our proposed method, any hypothesis for f is directly derived from Z, every measured point, whether correctly or falsely detected, corresponds to a cell in hypothesis f_k. Therefore the computation of $P(Z_k|f_k)$ simplifies to counting the number of cells in f_k which are associated (thus correctly detected) or un-associated (thus a false alarm)

$$P(Z_k|\ f_k) = \lambda_b \cdot (Detect) \cdot (FalseAlarm) \tag{8.13}$$

with the details of (8.12) and (8.13) in place, (8.11) becomes

$$\begin{aligned} P(f_k|\ Z_{1:k},\ f_{k-1}) = \lambda_k \quad & \cdot \left[\prod_{j\in R\cup S} P_{Dyn}(f_{k,j}|\ f_{k-1,l_k^j})\right] \\ & \cdot \left[\prod_{j\in S} P_{Split}(s_{k-1,l_k^j})\right] \\ & \cdot \left[\prod_{j\in R\cup S} P_{Sep}(\{f_{k,j}\}, j \in R\cup S)\right] \\ & \cdot \left[\prod_{j\in R\cup S} P_{Detect}\right] \cdot \left[\prod_{j\in U} P_{False}\right] \end{aligned} \tag{8.14}$$

where j indexes the cells in one of the R, S and U sets containing mature cells, divided cells and false alarms respectively. P_{Detect} is the probability of cell detection which is a constant, set to 0.9 empirically based on the performance of the cell model in Sec. 6.4. P_{Dyn} represents the cell motion dynamics to assess a hypothetical cell j in frame k based on its location at time $k-1$. Based on hand-tracked cell motion, a constrained Gaussian random walk well-approximates the observed cell dynamics. Because there is no explicit prediction of the statistics from $k-1$ to k, rather an assessment of a given set of hypotheses at frame k relative to $k-1$, therefore nonlinear dynamics can readily be accommodated.

Stem cells can split and bear new cells. A typical stem cell must be mature before it can split, i.e., there is an age constraint based on which the likelihood of cell division can be estimated. Cell divisions associated to young stem cells, below some minimum age, are considered unlikely and will be penalized by P_{Split}. After the cell's age passes the minimum age constraint, the probability of cell-split will increase with increasing cell age. Thus P_{Split} predicts the cell division in frame k based on the cell age in frame $k-1$.

P_{False} is a penalty for the association of false alarms: Any hypothesis containing extra measurements which are not associated to a cell is unlikely and so should be penalized accordingly. Finally stem cells in our experiments can not overlap, hence the centres

of two nearby cells cannot be closer than the sum of their radii. P_{Sep} is the cell-distance penalty based on the cell center separation, and sets to zero the likelihood of any hypothesis containing inadmissibly close cell centers. As was discussed after (8.12), because any hypothesis for f is directly derived from Z, and because the inference of Z (Sec. 6.4) does not allow the creation of measurements closer than a specified minimum separation, therefore the P_{Sep} term is, in practice, only a formality.

Having the association problem specified by (8.14), the optimization problem is to find the best estimate among all possible hypotheses evaluated. We propose to use an extended version of the Hungarian method (E-Hungarian), discussed next.

8.1.4 The Hungarian Method

The Hungarian method as a primal-dual algorithm belongs to the class of linear programming methods which have been used for the assignment problem [95]. The basis of the Hungarian method was introduced by Egervary and Konig and it has been completed later by Kuhn [76]. Primal-dual algorithms are characterized by

- A primal vector and a dual feasible solution is maintained by the algorithm.
- One of the following tasks is performed by the algorithm in each iteration
    1. The primal vector is kept fixed and the dual feasible solution is changed.
    2. The dual solution is kept fixed and the primal vector is changed toward primal feasibility while satisfying the present dual solution.
- By iterating the algorithm, the primal vector progresses toward primal feasibility.

We wish to solve the tracking problem represented by matrix $\mathtt{F} = (\mathtt{F}_{jm})$ given the cost matrix $d = (d_{jm})$. Each element of the cost matrix represents the cost of associating

measurement m to cell j. To solve the assignment problem we need to minimize

$$A(\mathtt{F}) = \sum_{j=1}^{J_k} \sum_{m=1}^{M_k} d_{jm} \mathtt{F}_{jm} \tag{8.15}$$

subject to

1. $\sum_{m=1}^{M_k} \mathtt{F}_{jm} = 1 \quad \forall j \in [1, J_k]$
   Each row of $\mathtt{F}$ sums to one,

2. $\sum_{j=1}^{J_k} \mathtt{F}_{jm} = 1 \quad \forall m \in [1, M_k]$
   Each column of $\mathtt{F}$ sums to one,

3. $\mathtt{F}_{jm} \geq = 0 \quad \forall j \in [1, J_k]$ & $m \in [1, M_k]$
   $\mathtt{F}$ is non-negative,

4. $\mathtt{F}_{jm} = 0 \;\; or \;\; 1 \quad \forall j \in [1, J_k]$ & $m \in [1, M_k]$
   Each element of $\mathtt{F}$ is 0 or 1.

Each feasible solution of (8.15) is an assignment problem of order $J_k \times M_k$. The dual of (8.15) is to find $\delta = (\delta_1, \delta_2, \cdots, \delta_{J_k})$ and $g = (g_1, g_2, \ldots, g_{M_k})$ such that

$$B(\delta, g) = \sum_{j=1}^{J_k} \delta_j + \sum_{m=1}^{M_k} g_m \tag{8.16}$$

is maximized, subject to

$$\delta_j + g_m \leq d_{jm}, \quad j \in [1, J_k] \;\&\; m \in [1, M_k] \tag{8.17}$$

The constraint in (8.17) can be rewritten as

$$\bar{d}_{jm} = d_{jm} - \delta_j - g_m \geq 0, \quad j \in [1, J_k] \;\&\; m \in [1, M_k] \tag{8.18}$$

which is called the dual feasibility condition for (δ, g), where $\bar{d} = (\bar{d}_{jm})$ is the reduced cost matrix and its elements $\bar{d}_{jm}$ are the reduced cost coefficients. As a result the vectors (δ, g) are dual feasible if and only if the reduced cost matrix $\bar{d} \geq 0$ [95]. An assignment problem $\mathtt{F}$ and a dual feasible solution (δ, g) are optimal if

$$\mathtt{F}_{jm}(d_{jm} - \delta_j - g_m) = \mathtt{F}_{jm}\bar{d}_{jm} = 0, \quad j \in [1, J_k] \;\&\; m \in [1, M_k] \tag{8.19}$$

which is called the complementary slackness optimality condition for the assignment problem and its dual. These are the basic assumptions in the Hungarian method such that it begins with a dual feasible solution and tries to find an assignment with allocations among the cost matrix elements which satisfy

$$\bar{d}_{jm} = d_{jm} - \delta_j - g_m = 0 \tag{8.20}$$

These elements of d are known as admissible elements. To update the cost matrix after each dual solution change in each stage, reduced cost coefficients are computed once, thus computational complexity per stage is at most $\bigcirc(n^2)$. As a result the overall computational complexity of the Hungarian method with consideration of n stages will be $\bigcirc(n^3)$ [95].

8.1.5 Extended Hungarian JPDA

The tracking problem can be represented in the form of an assignment problem so that a primal-dual algorithm can be applied to solve it. To represent the tracking problem in the form of an assignment matrix, the measurements Z_k of frame k (hypothesized cell locations) are assigned to the estimated cells $\hat{f}_{k-1}$ of frame $k-1$, giving rise to an assignment matrix (see Fig. 8.3) which represents the association of measurement $m \in [1, M]$ to cell $j \in [1, J]$.

		Meas.			Dummy Meas.					
		A	B	C	D	D	D	D	D	D
Cells	1	0	0	0	0	0	0	0	1	0
	2	0	0	1	0	0	0	0	0	0
	3	0	0	0	1	0	0	0	0	0
Cell Copies	$\bar{1}$	0	0	0	0	0	1	0	0	0
	$\bar{2}$	1	0	0	0	0	0	0	0	0
	$\bar{3}$	0	0	0	0	1	0	0	0	0
Dummy Cells	D_c	0	0	0	0	0	0	0	0	1
	D_c	0	1	0	0	0	0	0	0	0
	D_c	0	0	0	0	0	0	1	0	0

Figure 8.4: Tracking matrix illustrating splitting and un-association: In this example measurements A and C are both assigned to cell 2 (showed by rows 2 and $\bar{2}$), measurement B is assigned to a dummy cell. The conclusions are that cell 2 has split, cells 1 and 3 are undetected, and measurement B is a false alarm.

Although the Hungarian method efficiently and optimally solves single-frame assignment problems, it is insufficiently general to solve the cell tracking problem of interest because the one-to-one assignment precludes cell division and false alarms. In particular, recall the simple example of Fig. 8.3 in which the measurements A, B and C in the current frame are assigned respectively to the cells $1, 3$ and 2 in the previous frame. To allow one of the cells $(1, 2, 3)$ to split or one of the measurements (A, B, C) to be false requires a different approach. The key concept of this paper is that cell splits and false alarms can still be accommodated in a one-to-one assignment problem by enlarging the assignment matrix.

In our proposed Extended Hungarian (E-Hungarian) method, we double the number of rows to allow cell splitting and then add extra rows with the same number of measurements to allow false alarms. To allow mis-detection and to make the tracking matrix square, new columns are added. In this way the copied rows ($\bar{1}, \bar{2}$ and $\bar{3}$ in the example in Fig. 8.4) allow cell splitting so that by duplicating each row up to two measurements in the current frame can be associated to the same cell in the previous frame (represented by the rows) while each measurement assigned to the extra rows (dummy cells) will be interpreted as false alarms. Similarly the extra columns represent dummy measurements; any assignment from cell rows to these columns will be considered mis-detection, implying that one of the cells in frame $k-1$ is not associated to frame k.

To solve the cell association problem by the proposed method, we embed the proposed NNJPDA (8.14) as cost function d in the E-Hungarian method, so that each element of which (d_{jm}) represents the cost of assigning the measurement m in time k to the target j from time $k-1$. The goal is minimizing the cost of joint association of targets to measurements. To derive the cost matrix d from (8.14) we compute the following:

1. $Dyn_{jm} = \left[\frac{1}{P_{Dyn}(z_k^m |\ u_{k-1}^j)}\right]$

2. $Split_{jm} = \left[\frac{1}{P_{Split}(s_{k-1}^j)^2}\right]$

3. $Detect = \left[\frac{1}{P_{Detect}}\right]$

4. $False = \left[\frac{1}{P_{False}}\right]$

Then each element d_{jm} of the cost function d is obtained by

$$d_{jm} = \begin{cases} Dyn_{jm} \times Detect \times False \\ \qquad\qquad\qquad j \in \{Cells\} \\ Dyn_{jm} \times Split_{jm} \times Detect \times False \\ \qquad\qquad\qquad j \in \{Cell\ Copies\} \\ P_{Unassociation} \\ \qquad\qquad\qquad j \in \{Dummy\ Cells\} \\ P_{Unassociation} \\ \qquad\qquad\qquad m \in \{Dummy\ Meas.\} \end{cases} \tag{8.21}$$

where we note that the P_{Sep} term of (8.14) does not appear, having been satisfied inherently by the measuring process. Finally having d as the cost function, the proposed E-Hungarian-JPDA finds the optimal assignment by satisfying (8.19).

By minimizing the cost function d we are maximizing the Non-linear Non-Gaussian Joint Probabilistic Data Association in (8.14). Therefore the optimal association among all possible hypotheses is found by employing E-Hungarian method to solve (8.14) and the exact solution for frame k given $(k-1)$ is obtained.

8.2 Results

We have measured the performance of the proposed method and compared it with nearest neighbour (NN) [31] and standard JPDA [114]. Because of the difficulty in obtaining association ground truth for laboratory videos and because we wish to distinguish between errors in cell detection and cell association, therefore synthetic video clips simulating the random behaviour of stem cells were generated, in which non-overlapping cell dynamics were applied. We generated 1400 video clips, each video clip composed of 50 frames and

5 cells. The cells do not split to allow methods such as NN and standard JPDA, which do not support splitting, to be tested. We propose to assess the performance of the algorithm based on the average percentage of frames in which a cell has been correctly associated in comparison with ground truth.

The performance of the proposed method in comparison with NN is depicted in Fig. 8.5. The two methods are compared for different values of the probability of detection (P_{Detect}). For each value of P_{Detect}, 200 video clips, each composed of 50 frames, are generated and the synthetic cell centres are tracked over time applying the proposed method and NN. As we can observe, the proposed method has outperformed NN for all values of P_{Detect}. A comparison of the two algorithms for $P_{Detect} = 100\%$, $P_{Detect} = 95\%$ and $P_{Detect} = 90\%$ is depicted in greater detail, sequence by sequence, in Figs. 8.6, 8.7, and 8.8 respectively.

Fig. 8.9 plots the probability of perfect tracking of a cell over all 50 frames. The average tracking probability is computed over 200 video clips for each value of P_{Detect} (overall 1400 video clips composed of 50 frames). As can be seen the proposed method performs much better than NN, especially for $P_{Detect} > 90\%$.

Fig. 8.10 shows the performance of the proposed method in comparison with the standard JPDA for $P_{Detect} = 100\%$. To compare the results, the performance of JPDA is measured as a function of gate area $G_v = \pi \times G_r^2$, where $G_r = G_f \times \sigma_{rw}$ is the gate radius, set to be a multiple (G_f) of the standard deviation of the constrained Gaussian random walk. Fig. 8.10 clearly shows that increasing G_f improves the performance of JPDA towards that of the E-Hungarian-JPDA, which is expected since for a sufficiently large G_f JPDA is testing all hypotheses. For $G_f = 4$, where the performance of the two methods is almost equal, the gating area is a circle with diameter of $2 \cdot G_r = 20$ pixels. Considering that each well is about 70×50 pixels, such a circular gate covers a significant fraction of the well's area around each cell centre. As the circular gate completely covers the

field of measurements, standard JPDA evaluates all possible $n!$ hypotheses to find the best association to assign n sources to n sinks, much more expensive than E-Hungarian-JPDA.

To generate results using real data, we have applied the proposed method to long streams of microscopic phase contrast HSC video. First, the cell center candidates are located by applying the probabilistic cell model in Sec. 6.4 in which the cell candidates are found by locating and thresholding the local maxima in a cell probability map. Then, to track the cells over time, our proposed E-Hungarian-JPDA method has been applied to the localized cell centers as potential HSC candidates.

Fig. 8.11 shows the detected non-dividing cell centers in 15 frames of a HSC video clip spanning 45 minutes of time (successive frames 3 minutes apart). The proposed E-Hungarian-JPDA method is applied to the detected cell centres, with the tracking results depicted in Fig. 8.12. As can be observed in Fig. 8.12, the proposed E-Hungarian-JPDA method is able to associate the non-dividing HSCs correctly.

The proposed method is also capable of tracking more challenging *dividing* HSCs, which is not the case for standard JPDA. Fig. 8.13 again shows the detected dividing cell centers in 15 frames of HSC video. The association results obtained by applying the proposed tracking method are depicted in Fig. 8.14. Again the dividing HSCs are tracked correctly by the E-Hungarian-JPDA method.

To see the tracking of splitting cells in greater detail, Figs. 8.15, 8.16, and 8.17 show the details of two dividing stem cells over time.

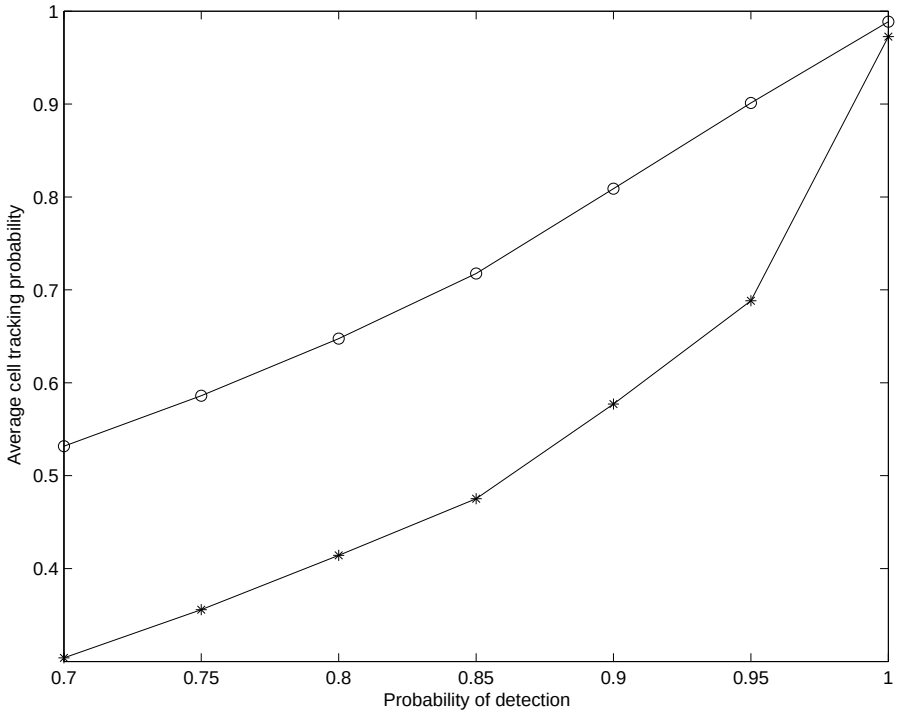


Figure 8.5: Tracking Performance: E-Hungarian-JPDA(o) vs. Nearest Neighbour(*) as a function of the probability of detection (P_{Detect}). For each value of probability of detection, 200 video clips each of 50 frames have been used.

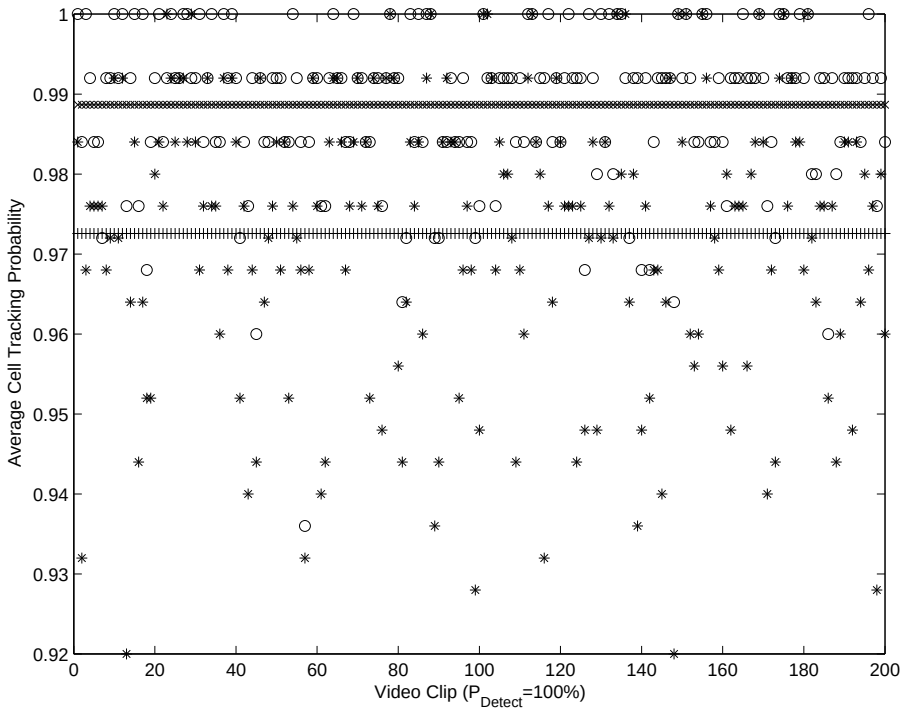


Figure 8.6: Tracking performance of E-Hungarian-JPDA(o) vs. Nearest Neighbour(*) for perfect detection ($P_{Detect} = 100\%$) where the average probabilities over 200 video clips are superimposed.

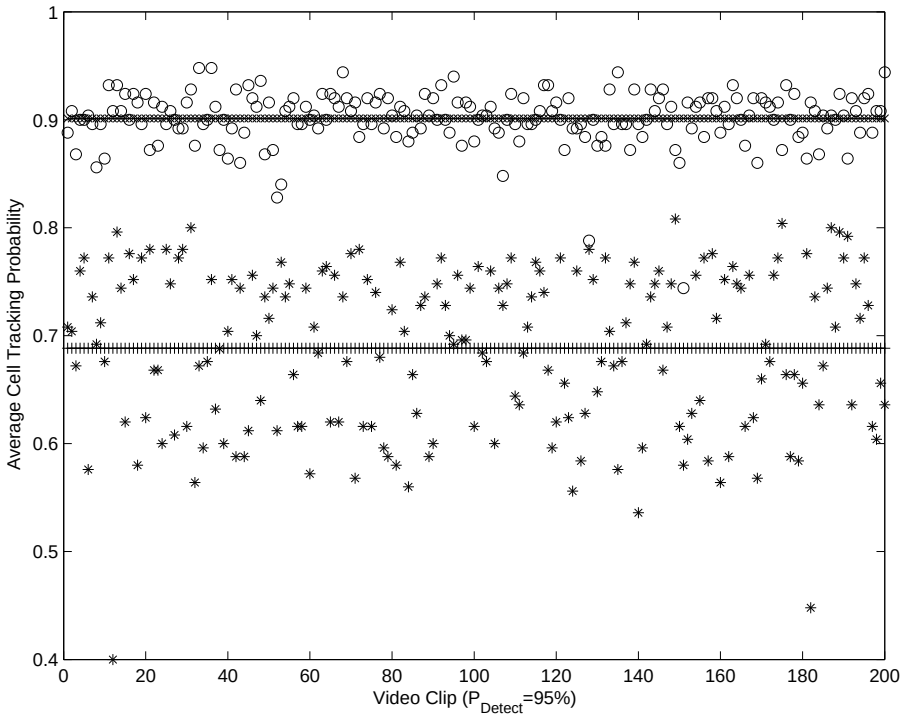


Figure 8.7: Tracking performance of E-Hungarian-JPDA(o) vs. Nearest Neighbour(*) for $P_{Detect} = 95\%$ where the average probabilities over 200 video clips are superimposed.

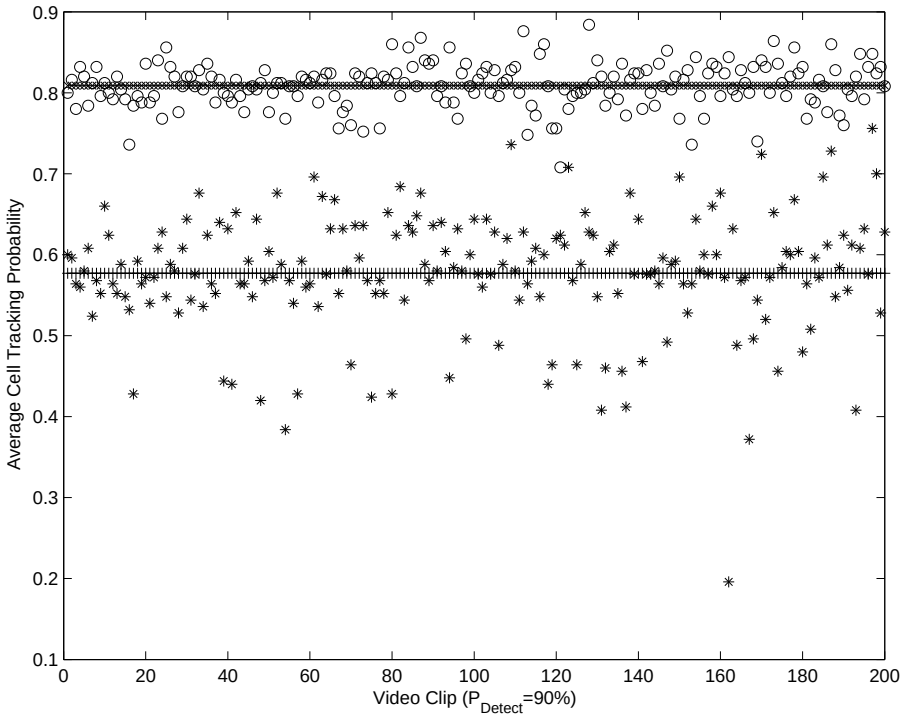


Figure 8.8: Tracking performance of E-Hungarian-JPDA(o) vs. Nearest Neighbour(*) for $P_{Detect} = 90\%$ where the average probabilities over 200 video clips are superimposed.

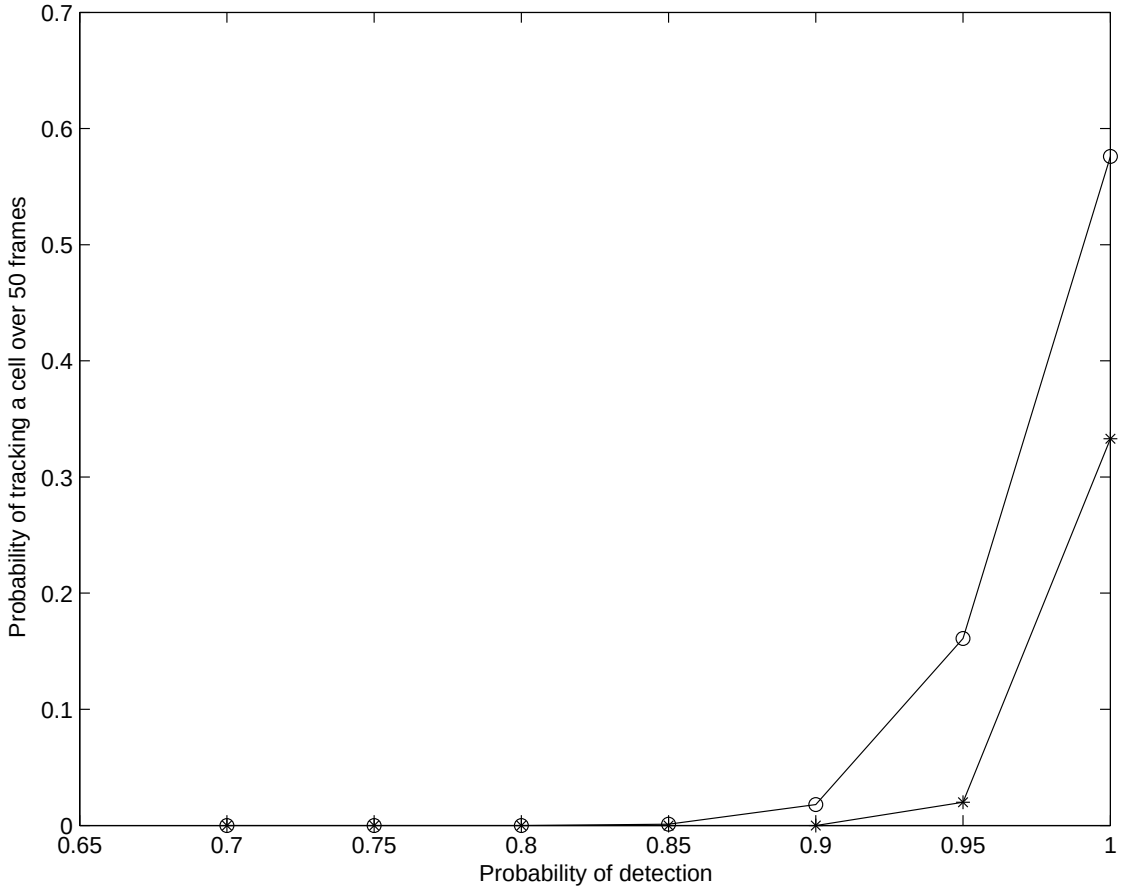


Figure 8.9: Probability of perfect tracking in which a typical cell is tracked over all frames from the first to the last frame: E-Hungarian-JPDA(o) vs. Nearest Neighbour(*) as a function of the probability of detection (P_{Detect}). In each case 200 video clips each of 50 frames are used.

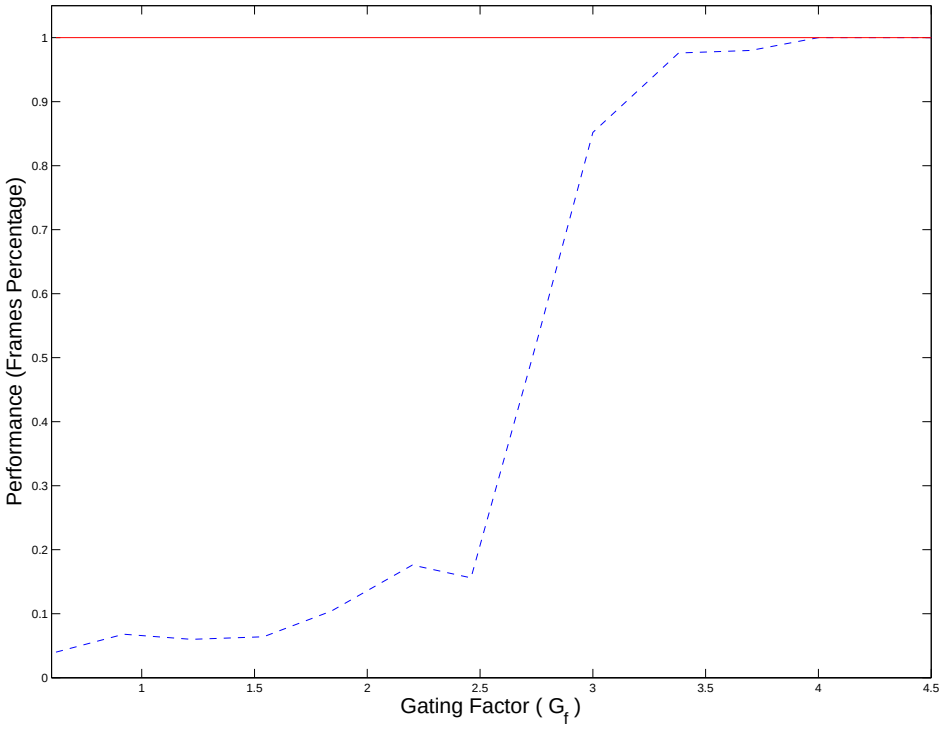


Figure 8.10: Performance of the proposed E-Hungarian-JPDA (solid line) in comparison with standard JPDA (dashed line) as a function of gating factor (G_f).

Image Sequence Without Cell Division

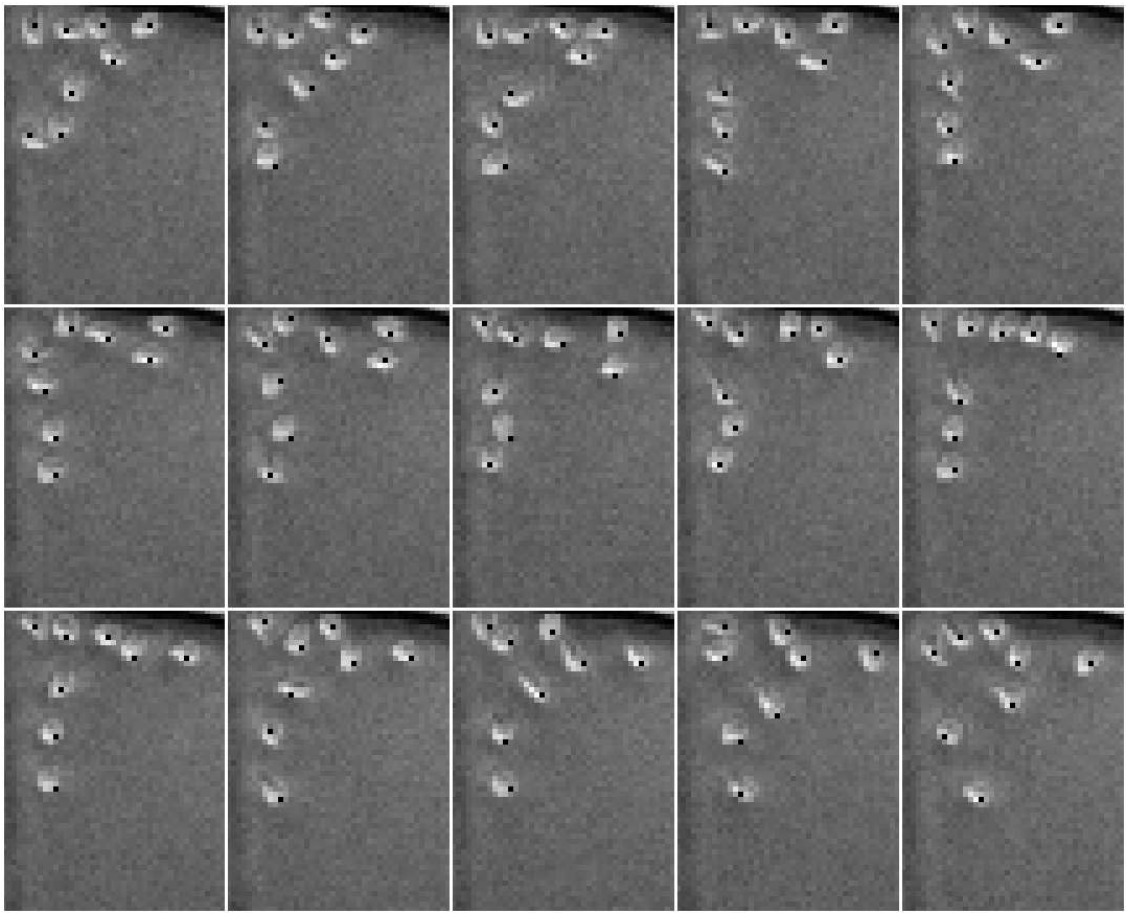


Figure 8.11: An example illustrating the detection by applying the proposed cell model in Sec. 6.4. The detection results are superimposed on the original HSCs as black dots locating the detected cell centres.

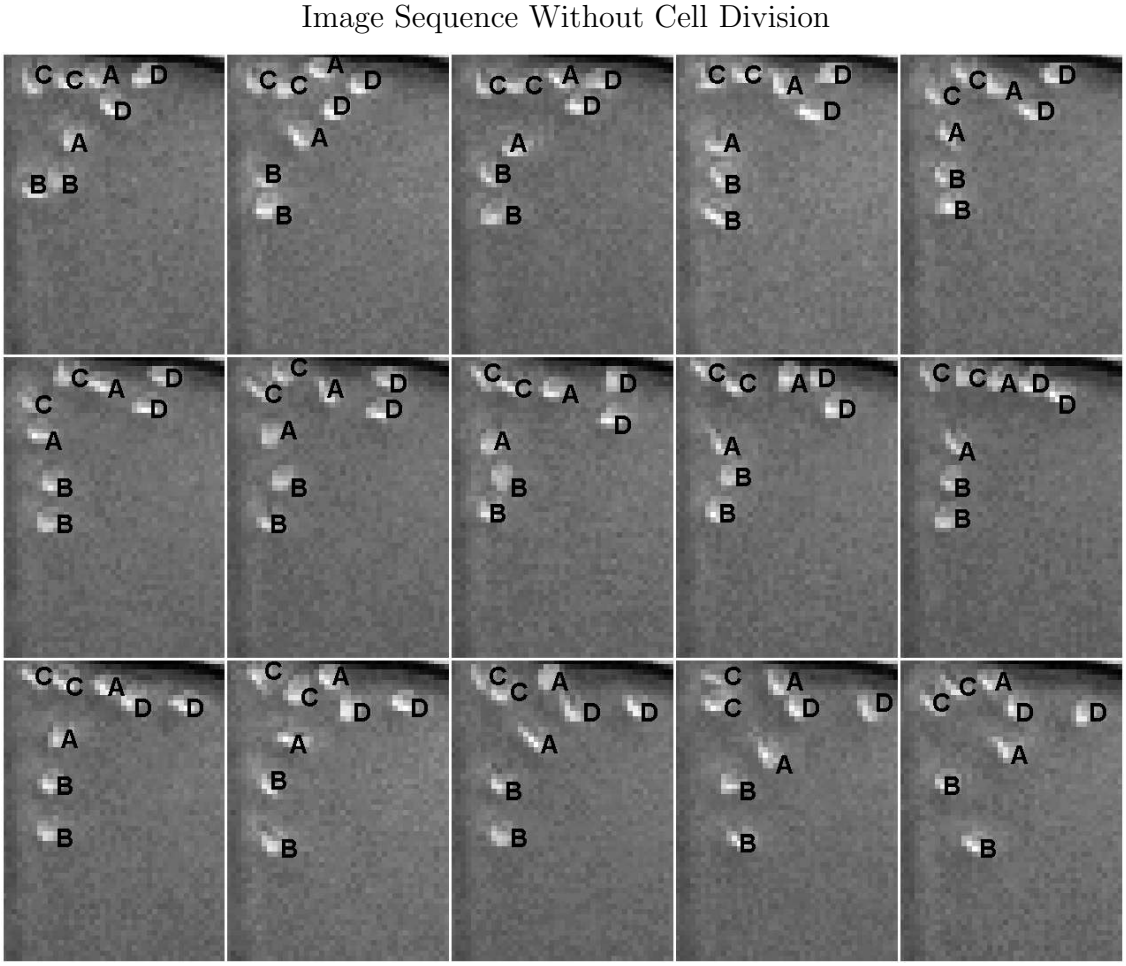


Figure 8.12: An example illustrating the association by applying the proposed E-Hungarian-JPDA respectively. The association results are superimposed on the original HSCs such that each letter shows a different cell track/split over time.

Image Sequence With Cell Division

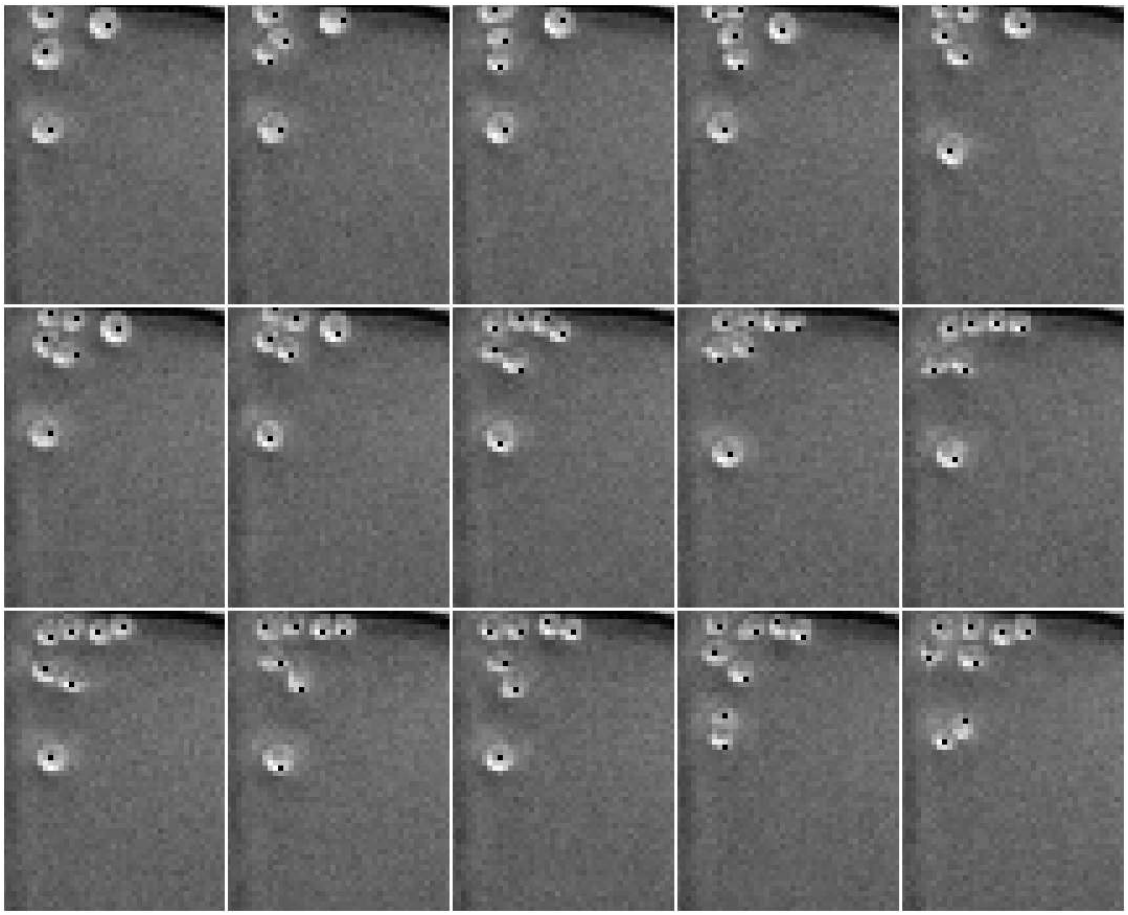


Figure 8.13: An example illustrating the detection by applying the proposed cell model in Sec. 6.4. The detection results are superimposed on the original HSCs as black dots locating the detected cell centres.

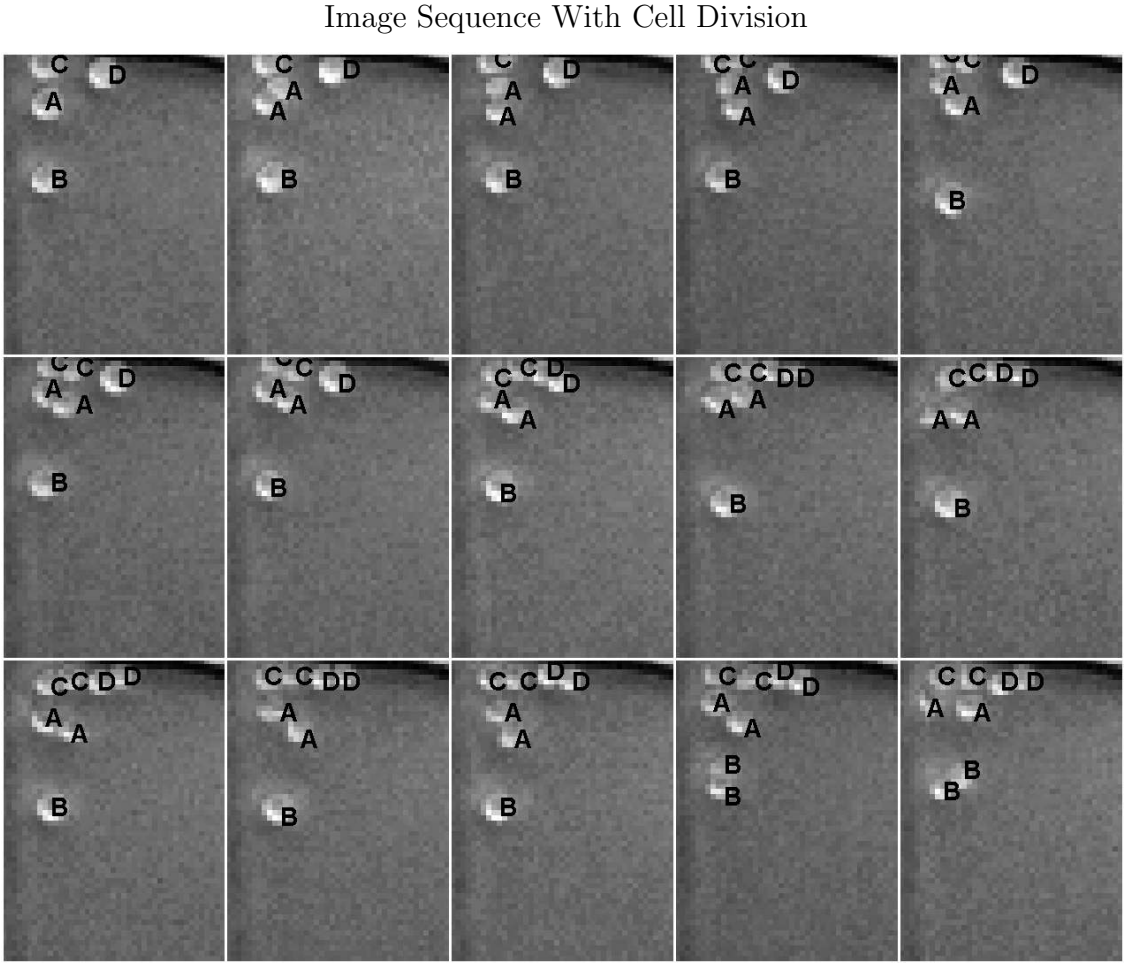


Figure 8.14: An example illustrating the association by applying the proposed E-Hungarian-JPDA. The association results are superimposed on the original HSCs such that each letter shows a different cell track/split over time.

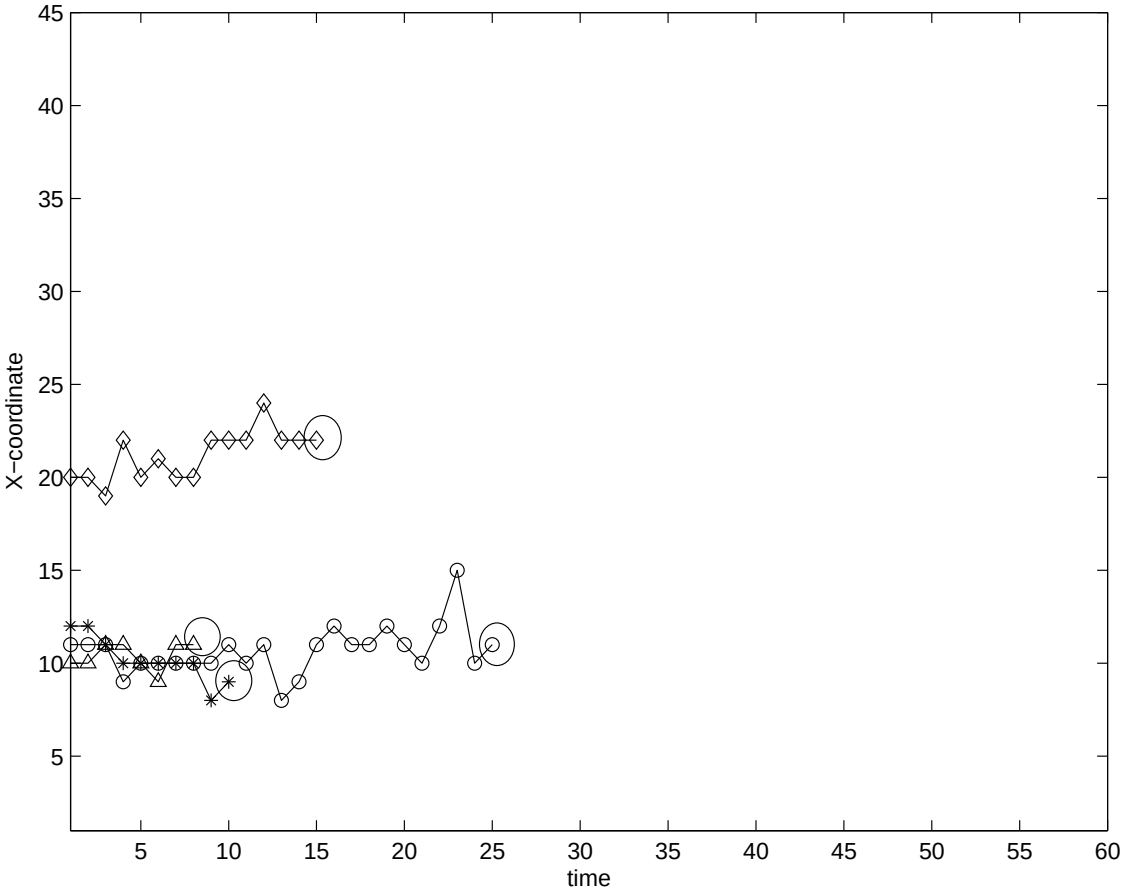


Figure 8.15: Blood stem cell tracks over time: Tracks of four mature cells from Fig. 8.14; the large circle at the end of each track highlights the division point over time when a mature cell divides to produce two new cells.

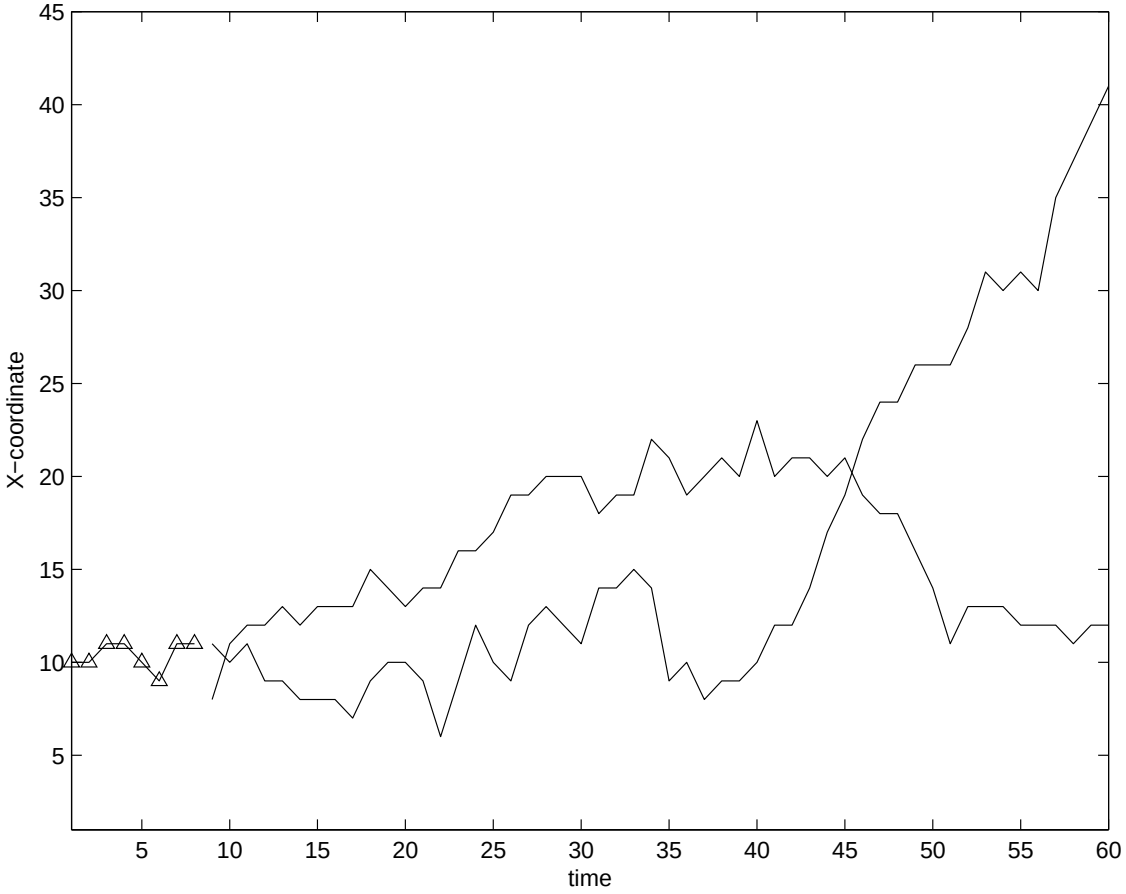


Figure 8.16: Blood stem cell tracks over time: The triangled line represents the track of the mature cell from Fig. 8.15, before splitting, then dividing into two new tracks (solid lines).

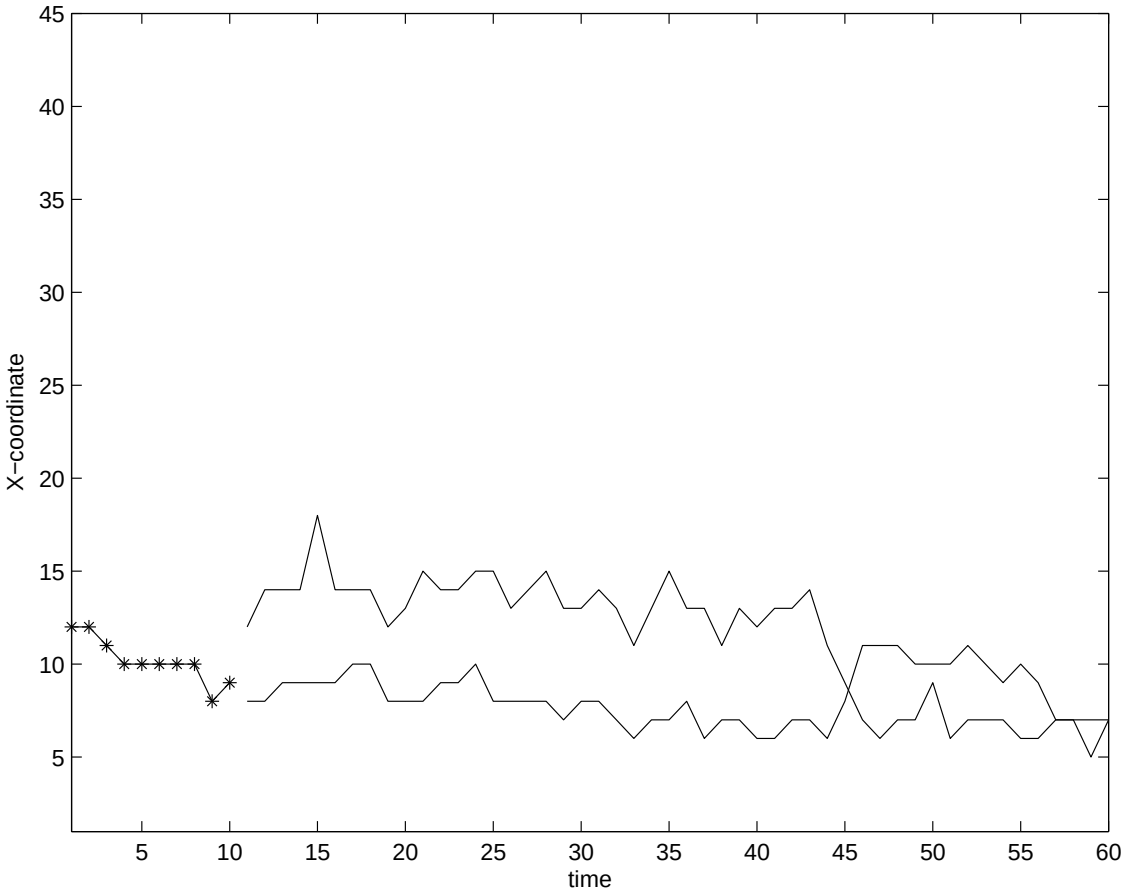


Figure 8.17: Blood stem cell tracks over time: The starred line represents the track of the mature cell from Fig. 8.15, beginning as a mature cell and dividing into two.

Chapter 9

Conclusions and Future Work

Image cytometry is a practical approach to measure and extract cell properties from large volumes of microscopic cell images. As an important application of image cytometry, this thesis presents a probabilistic model-based cell tracking method to locate and associate HSCs in phase contrast microscopic images.

In this research, a model-based cell tracker is designed to locate and track individual HSCs in three dimensional (3-D) cell image sequences that contain two spatial and one temporal dimensions. The proposed cell tracking system consists of three inter-related stages:

- Cell detection/localization,
- The association of detected cells,
- Background estimation/subtraction.

The key ideas and contributions of this research can be summarized as

- The design of model-based object localization and/or segmentation methods,

- Present object localization as an inverse problem and provide two deconvolution based solutions,
- Develop a mutual method for foreground localization and background estimation,
- The introduction of an optimal generative single-frame association for multi-target tracking based on an extended Hungarian method which is a class of linear programming optimization,
- Ridgelet BayesShrink for image denoising.

Model-Based Cell Detection - Chapter 6

After carefully observing HSCs in typical image sequences, a probabilistic model-based method was proposed in Sec. 6.2 to detect HSCs of Phenotype 1 by capturing the key properties of these cells as product of the cell boundary P_{cb}, cell interior P_{ci} and boundary uniformity P_{bu} terms.

The cell model in Sec. 6.2 was generalized to be applied to different HSC phenotypes investigated in this work. The improved cell model discussed in Sec. 6.4 combines the common properties of HSC Phenotype 1, 2, and 3 in a single model. The proposed probabilistic cell model in Sec. 6.4, models a cell by the product of the cell probability based on cell interior, the penalizing probability based on outer cell ring, and the discrimination probability based on pixel pairs located on opposite sides of cell center.

Present The Object Detection as a Deconvolution Problem - Chapter 6

Cell centre localization is essentially an inverse problem which can be addressed in the form of a deconvolution problem. Considering cell detection as an inverse problem, two approaches are proposed as the potential solutions in the form of a deconvolution problem.

The first approach can be summarized as follows:

- Cell areas containing individual or groups of cells are segmented first,
- The cell center localization is solved by finding a set of cell shape parameters for optimal representation of cell segmented areas using an optimized ellipse fitting method.

This method is a generic method, capable of modelling different cell types by designing the proper shape model. The optimized cell shape parameters can be then extracted using the same search method as the proposed method by optimizing the cost function that fits the cells in the segmented area.

The proposed method effectively model splitting and close by cells, and is capable of modelling different cell types with changes in the model parameters. However, in cases where, either a complex parameterized shape is needed to model a cell, or an exact cell segmentation is in demand in place of cell center localization, this method will not be effective.

In contrast, in the second approach the inverse problem is solved as follows:

- Cell centers are located first,
- Cell segmentation will be solved by finding cell regions for optimal representation of cell centers.

The proposed method consists of cell template generation, template matching, cell center localization, and watershed segmentation.

Mutual Foreground Detection and Background Estimation - Chapter 7

A novel mutual algorithm for object detection and background estimation is proposed. The proposed background estimation method employs cell detection to remove the foreground objects and estimates the background over the 3-D residual sequence.

The key contributions of the proposed method are first, that the proposed method addresses foreground segmentation and background estimation as inter-related processes, and takes advantage of this inter-relation to improve the performance of each process by integrating the outcome of another reciprocally. The proposed probabilistic model-based localization algorithm will be applied to the image sequence to localize cell centers. Localized cells, the foreground elements, are then removed from the scene frame by frame. The resultant residual image sequence is then used for an accurate spatio-temporal background estimation.

Second, the proposed method is an adaptive method in comparison with other methods which are either point-wise or window-wise (including texture-based methods). In this way, as long as there is sufficient information in the 3-D residual sequence the proposed method takes advantage of point-wise estimation methods to estimate the background. However if an accurate estimation of background pixel is not possible, due to the lack of temporal information, the algorithm extends adaptively over a multi-clique neighbourhood until a precise estimation of background is possible. The proposed method switches between point-wise and window-wise as necessary.

Introduce an Optimal Single-Frame Association for Multi-Target Tracking - Chapter 8

The proposed method in chapter 8 presents an optimal single-frame assignment solution for object association that has been applied to track HSCs in phase contrast microscopic images. The proposed approach uses linear programming optimization, based on an extended Hungarian method. This is a generative algorithm and can be used with various tracking methods, including nearest-neighbor, PDA, JPDA, particle filtering, MHT and deformable models, by designing the correct cost function.

The contribution of the proposed tracking method is a generalization of the Hungarian method to allow association with track divisions, false alarms, and missed detections. We are motivated to consider alternative generalizations for other classes of tracking problems.

Ridgelet BayesShrink for Image Denoising - Chapter 5

In the proposed method in chapter 5, the BayesShrink ridgelet is introduced for image denoising. The performance of the proposed method is compared with that of the VisuShrink ridgelet image denoising method. The experimental results by the proposed method shows the superiority of the image quality and its higher SNR in comparison with VisuShrink ridgelet technique.

To further improve the performance of wavelet method for denoising edges and lines, the ridgelet based method is combined with its wavelet counterpart to construct a combined method that performs better than the application of either method individually.

Final Word

As it was mentioned in Chapter 6, each detection method has some advantages, some limitations and shortcomings. Applicability and advantages of each method were discussed. Based on specific data set one or several detection methods might be applicable. The initial cell model in Chapter 6, for example, performs equally well as the thresholding detection method and general probabilistic model, however this model is specifically designed based on attributes of regular HSCs and performs very poorly where is applied to the other cell data.

Cell detection responds robustly for the noise level with *PSNR* greater than 15. However, higher noise levels are tolerated while background estimation/subtraction method discussed in Chapter 7 is applied first. Moreover, application of the second phase of back-

ground estimation/subtraction improves the detection performance at least by 10%.

As it was discussed in Chapter 8, the proposed E-Hungarian-JPDA method is able to associate both non-dividing and more challenging dividing HSCs correctly. However, we should point out that cell detection, background estimation, and cell association are inter-related, and in turn affect the performance of association, so that even an optimal association approach like the proposed method cannot compensate and recover from poor detection.

Future Work

To extend the proposed methods in this research the potential directions can be summarized as follows.

As the proposed optimal single frame approach is a generative algorithm, it can be used with various tracking methods and there is potential motivations to consider alternative generalizations for other classes of tracking problems. Future work will be conducted to extend the proposed model-based tracking system by integrating the information over multiple neighboring frames.

Future work can also focus to design an iterative version of the proposed foreground detection and background estimation method. In such an extension, the segmentation and estimation results will iteratively be used to improve the performance of each other.

As the proposed model-based tracking is a general framework, the proposed approach can be potentially extended to be applied to 4-D tracking applications consisting of 3-D spatial and 1-D temporal dimensions.

Further, potential extensions of this research would be two-dimensional, three-dimensional, and four-dimensional segmentation, localization and multi-target tracking with a broad range of applications in biomedical research including non-invasive surgery, remote/robot

surgery, stem cell research/therapy, tissue engineering and automated diagnostic systems.

Future work will be conducted to integrate detection/segmentation and tracking in a single Bayesian model. The proposed future work would be a general framework to solve association-based object tracking. In such a general model object detection and data association will mutually be integrated to be solved together. An object detection algorithm such as the probabilistic model-based approach that was proposed in Sec. 6.4 can be employed for detection. Detected objects are further tracked by applying an association method such as the one that was proposed in chapter 8. In this approach detection and association stages mutually resolve ambiguities. Moreover, this is a generative framework and a variety of different detection and association methods can be used in the proposed future work based on the application at hand. The proposed future work is discussed with more detail in Appendix 1.

Appendix A

The Proposed Future Work

The proposed future work, a Bayesian model for mutual detection and tracking is discussed here.

A.1 A Bayesian Framework for Mutual Detection and Tracking

As it was discussed in chapter 8, the optimal solution for a tracking problem is the *maximum a posteriori* estimation of $F_{1:K}$:

$$\hat{F}_{1:K} = arg \max_{F_{1:K}} P(F_{1:K}|\ I_{1:K}, f_0) \tag{A.1}$$

From Bayes' rule,

$$\begin{aligned} P(F_{1:K},\ I_{1:K}, f_0) &= P(F_{1:K}|\ I_{1:K}, f_0)P(I_{1:K}, f_0) \\ &= P(I_{1:K}|\ F_{1:K}, f_0)P(F_{1:K}, f_0) \end{aligned} \tag{A.2}$$

As $P(I_{1:K}, f_0)$ is fixed, $\hat{F}_{1:K}$ doesn't depend on it, thus

$$P(F_{1:K}|\ I_{1:K}, f_0) \propto P(I_{1:K}|\ F_{1:K}, f_0)P(F_{1:K},\ f_0)$$

At the same time $P(f_0)$ is fixed

$$P(F_{1:K},\ f_0) = P(F_{1:K}|\ f_0)P(f_0) \propto P(F_{1:K}|\ f_0) \tag{A.3}$$

so we conclude that

$$\hat{F}_{1:K} = arg\ \max_{F_{1:K}}\ P(I_{1:K}|F_{0:K}) \cdot P(F_{1:K}|f_0) \tag{A.4}$$

Since $F_{1:K} = \{f_1,\ f_2,\ ...,\ f_K\}$, the solution to (A.4) is realized, in principle, by examining and evaluating all possible object parameterizations and associations. In virtually all tracking problems of this kind the problem is made tractable by searching over a limited number of hypotheses

$$\{F^h_{1:K}|\ h = 1, 2, ...\} \tag{A.5}$$

such that we find the best member of this set

$$\hat{F}_{1:K} = F^{\hat{h}}_{1:K} \quad \text{where} \quad \hat{h} = arg\ \max_{h}\ P(I_{1:K}|\ F^h_{0:K}) \cdot P(F^h_{1:K}|f_0) \tag{A.6}$$

The original, optimal solution is found if it is included among the hypotheses, i.e., if

$$arg\ \max_{F_{1:K}}\ P(F_{1:K}|\ I_{1:K},\ f_0) \in \{F^h_{1:K}\} \tag{A.7}$$

The key, here, to efficiency is to minimize the number of hypothesis; the key to quality of estimation is finding the most likely hypothesis. As these goals are in opposition, we are left with a complexity/quality trade off.

A.1.1 Evaluation of $P(I_{1:K}|F^h_{1:K})$

The detection method evaluates the likelihood of an object, given an image. To solve the MAP problem we need to compute $P(I_{1:K}|\ F_{1:K})$, the likelihood of a given image sequence as a function of a specified object parametrization and association. Since f_k provides a complete parameterized description of I_k, conditioned on $F_{1:k}$, $I_{1:k}$ is Markov:

$$P(I_{1:K}|\ F_{1:K}) = \prod_{k\in[1,K]} P(I_k|f_k) \tag{A.8}$$

A specific object model describes only the likelihood of an object; it says nothing about groups of objets, nor does it provide any kind of prior on object parameters. It follows, then, that as long as zero-probability hypotheses are not created, then all remaining hypotheses $\{f^h_k\}$ are equally likely *a priori*. Because $P(I_k)$ is fixed, and moreover because all valid hypotheses are equally likely, such that $P(z_k)$ is constant, we can conclude

$$P(I_k|\ z_k) \propto P(z_k|\ I_k), \tag{A.9}$$

implying that the evaluation of $P(I_k|\ z_k)$ can follow from evaluating $P(z_k|\ I_k)$. For example, the proposed parametric cell model from chapter 6 may be employed for cell tracking applications to evaluate $P(I_{1:K}|F^h_{1:K})$. Thus the proposed cell model $P(z_k|\ I_k)$ will be applied to each image frame I and a two dimensional probability map will be generated and hypothesized cells are located at local maxima of this map.

After measurement hypotheses are generated from I, they must be evaluated. To compute $P(I|\ f)$ the objects in f are divided into two sets:

1. f_M: Those objects in f which are located within δD of a measurement.

2. $f_{\bar{M}}$: Those objects in f which are not within δD of a measurement.

A third set contains the unmatched measurement:

3. $\bar{f}_M$: Those measurements which are not within δD of any object in f.

f_M contains the successful matches, $f_{\bar{M}}$ and $\bar{f}_M$ the failed ones. The fit between I_k and f_k^h is thus quantified as

$$P(I_k|\ f_k^h) = \left[\prod_{j\in f_M} P(z_{k,j}|\ I_k)\right] \cdot \left[\prod_{i\in \bar{f}_M} (1 - P(z_{k,i}|\ I_k))\right] \cdot \left[\prod_{j\in f_{\bar{M}}} P(z_{k,j}|\ I_k)\right] \tag{A.10}$$

where $P(z_{k,j}|\ I_k)$ is the probability of the location of the j^{th} object in the state f_k^h for frame k, and $P(z_{k,i}|\ I_k)$ is the probability of the i^{th} measurement in frame k. Thus $P(I_k|\ f_k^h)$ is evaluated by applying (A.10).

A.1.2 Evaluation of $P(F_{1:K}^h|f_0)$

The second part of (A.6) is the evaluation of association hypotheses $\{F_{1:K}^h\}$. To track objects over time, detected objects in the measurement hypotheses of the current frame z_k must be associated to the most probable element in the previous frame.

Considering that for each image frame k, we associate objects features from $(k-1)$ only, Markovianity can be asserted on $F_{1:K}$ such that

$$P(F_{1:K}|f_0) = \prod_{k\in[1,K]} P(f_k|f_{k-1}) \tag{A.11}$$

where f_k is the set of object properties in frame k. Each object in f_k must belong to one of the Unassociated (U), New (N), and Regular (R) sets. In contrast with Joint Probabilistic Data Association (JPDA) [6, 31] in which new tracks can not be initiated, the proposed method initiates new tracks for new objects, therefore the following constraints are considered:

(i) Each measurement must originate from object or clutter.

(ii) Each measurement can be associated to one object.

(iii) An un-associated measurement in frame k can initiate a new track.

Asserting Markovianity we evaluate $P(f_k^h|\ f_{k-1}^h)$ in the rest of this section. The association problem is resolved frame by frame by selecting the hypothesis with the maximum joint association probability. In this way the measurement hypothesis z_k for frame k and association hypothesis f_{k-1}^h from the previous frame are used to generate hypotheses f_k^h. Therefore we have

$$P(f_k^h|\ f_{k-1}^h) = P(f_k^h|\ z_{1:k}, f_{k-1}^h) = P(f_k^h|\ z_k, f_{k-1}^h) \tag{A.12}$$

The filter step is

$$P(f_k^h|\ f_{k-1}^h) = P(f_k^h|\ z_{1:k}, f_{k-1}^h) = \frac{P(f_k^h|\ z_{1:k-1}) \cdot P(z_k|\ f_k^h)}{P(z_k|\ z_{1:k-1})} \tag{A.13}$$

$P(z_k|\ z_{1:k-1})$ is fixed and we have

$$P(f_k^h|\ f_{k-1}^h) = \lambda_k \cdot P(f_k^h|\ z_{1:k-1}) \cdot P(z_k|\ f_k^h) \tag{A.14}$$

where λ_k is a normalization constant. The first term of (A.14), $P(f_k^h|\ z_{1:k-1})$, is a prediction step which is illustrated as follows.

The prediction step can be explained in the proposed method as

$$P(f_k^h|\ z_{1:k-1}) = \int P(f_k^h|\ f_{k-1}^h) \quad P \quad (f_{k-1}^h|\ z_{1:k-1}) df_{k-1}^h = \tag{A.15}$$
$$\left[\prod_{j \in R \cup N} P_{vel}(z_{k,j}, z_{k-1,j})\right] \quad \cdot \quad \left[\prod_{j \in N} P_{state})\right]$$

The former term P_{vel} predicts the location of the hypothetical object j in frame k based on its dynamics and its location in frame $k-1$. The latter term P_{state} predicts the likelihood of presence of a new object in frame k.

The second term of (A.14), $P(z_k|\ f_k^h)$, is the likelihood of measurement z_k given hypothesis f_k^h and is given by

$$P(z_k|\ f_k^h) = \left[\prod_{j \in R \cup N} P_{pdf}(v_{k,j}, 0, C_{k,j})\right] \cdot \left[P_{sep}(f_k^h)\right] \cdot \left[\prod_{j \in U} P_{una}\right] \tag{A.16}$$

where $v_{k,j} = z_{k,i} - \hat{z}_{k,j}$ is an innovation term so that the i^{th} measurement is within δD of the j^{th} hypothesized object location in frame k. P_{una} is a penalty on the association of un-associated objects, and P_{sep} is the probability of separation distance of a measurement pair. As we can see in the proposed method, the likelihood of measurement $z_{k,j}$ is penalized by the unlikely events such as minimum separation distance and un-associated objects.

Bibliography

[1] A region-region and region-edge cooperative approachof image segmentation. In *Proc. of IEEE Intl. Conf. on Image Processing (ICIP)*, volume 3, page 470474, 1994.

[2] F. Aghili and M. Namvar. Adaptive control of manipulators using uncalibrated joint-torque sensing. *IEEE Transactions on Robotics*, 22(4):854–860, 2006.

[3] C. Andrieu, A. Doucet, S. Singh, and V. Tadic. Particle methods for change detection, system identification, and control. *Proceedings of the IEEE SMC*, 92(3):423–438, 2004.

[4] D. Anoraganingrum. Cell segmentation with median filter and mathematical morphology operation. In *IEEE International Conference on Image Analysis and Processing*, pages 1043–1046, 1999.

[5] D. H. Ballard. Generalized the hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):1111–122, 1981.

[6] Y. Bar-Shalom and K. Birmiwal. Consistency and robustness evolution of the pdaf for target trackingin a cluttered environment. *Automatica*, 19:431–437, 1983.

[7] Y. Bar-Shalom and E. Tse. Tracking in a cluttered environment with probabilistic data association. *Automatica*, 11:451–460, 1975.

[8] I. Bauman, R. Nenninger, H. Harms, H. Zwierzina, K.Wilms, A.C. Feller, V.T. Meulen, and H.K. Muller-Hermelink. Image analysis detects lineage-specific morphologic markers in leukemia blast cells. *American Journal of Clinical Pathology*, 105(1):23–30, 1995.

[9] P. Van Beek, A.M. Tekalp, N. Zhuang, I. Celasun, and Xia Minghui. Hierarchical 2-d mesh representation, tracking, and compression for object-based video. *IEEE Transactions on Circuits and Systems for Video Technology*, 9(2):353 – 369, 1999.

[10] S. Beucher and C. Lantuejoul. Use of watersheds in contour detection. *Proc. International Workshop on Image Processing, Real-Time Edge and Motion Detection/Estimation*, pages 17–21, 1979.

[11] S. Beucher and F. Meyer. *Mathematical Morphology in Image Processing, Chapter 12-The morphological approach to segmentation: the watershed transformation.* Marcel Dekker, New York, NY, 1993.

[12] W. Bieniecki. Oversegmentation avoidance in watershed-based algorithms for color images. In *IEEE International Conference on Modern Problems of Radio Engineering, Telecommunications and Computer Science*, pages 169–172, 2004.

[13] A. Blake and M. Isard. The condensation algorithm - conditional density propagation and applications to visual tracking. In *Advances in Neural Information Processing Systems 9, MIT Press*, pages 361–368, 1997.

[14] A. Bobick, J. Davis, S. Intille, F. Baird, L. Cambell, Y. Irinov, C. Pinhanez, and A. Wilson. Kidsroom: Action recognition in an interactive story environment. In *MIT Perceptual Computing Technical Report*, 1996.

[15] P. Bonnin, J. Blanc Talon, J. Hayot, and B. Zavidovique. A new edge point/region cooperative segmentation deduced from a 3d scene reconstruction application. *SPIE Applications of Digital Image Processing XII*, 1153:579– 591, 1989.

[16] V. Boutenko, T. Lebihen, and A. Perchant. Digital subtraction x-ray cardioangiography using a recursive background estimation algorithm. In *Proc. Society of Photo-Optical Instrumentation Engineers (SPIE) Conference, Medical Imaging*, volume 3338, pages 799–805, 1998.

[17] E. Campo and E. Jaffe E. Mantle cell lymphoma. *Arch. Pathology Lab. Med.*, 120(1):12–14, 1996.

[18] E. J. Candes. *Ridgelets: Theory and Applications*. Ph.D. thesis, Department of Statistics, Stanford University, 1998.

[19] E. J. Candes and D. L. Donoho. Ridgelets: a key to higher dimensional intermittency? *Phil. Transactions Royal Society London*, pages 2495–2509, 1999.

[20] T. Carron and P. Lambert. Color edge detector using jointly hue, saturation and intensity. In *Proc. of 1994 Int. Conf. on Image Processing (ICIP)*, volume 3, pages 977–981, 1994.

[21] K. R. Castleman. *Digital Image Processing*. Prentice Hall, Englewood, New Jersey, 1996.

[22] S. G. Chang, B. Yu, and M. Vetterli. Adaptive wavelet thresholding for image denoising and compression. *IEEE Transactions on Image Processing*, 9(9):1532–1546, 2000.

[23] S. G. Chang, B. Yu, and M. Vetterli. Spatially adaptive wavelet thresholding with context modeling for image designing. *IEEE Transactions on Image Processing,*, 9(9):1522–1531, 2000.

[24] C. Chen. *Signal and Image Processing for Remote Sensing*. CRC, Taylor and Francis, USA, 2006.

[25] P. Chen and T. Pavlidis. Image segmentation as an estimation problem. In *Computer Vision Graphics and Image Processing*, volume 12, page 153 172, 1980.

[26] X. Chen, X. Zhou, and S.T.C. Wong. Automated segmentation, classification, and tracking of cancer cell nuclei in time-lapse microscopy. *IEEE Transactions on Biomedical Engineering*, 53(4):762–766, 2006.

[27] Y. Cheng. Mean shift, mode seeking, and clustering. *IEEE Trans. on Pattern Anal. Machine Intell.*, 17(8):790–799, 1995.

[28] E. Clarke, A.W. Wognum, R. Marciniak, and A.C. Eaves. Mesenchymal cell precursors from human bone marrow. *Blood*, 98:355–365, 2001.

[29] R. A. Close and J. S. Whiting. Motion-compensated signal and background estimation from coronary angiograms. In *Proc. Society of Photo-Optical Instrumentation Engineers (SPIE) Conference, Medical Imaging*, volume 2434, pages 185–194, 1995.

[30] R. Coifman and M. Wickerhauser. Entropy-based algorithms for best-basis selection. *IEEE Transactions on Information Theory*, 38(2):713–718, 1992.

[31] S. Colegrove, A. Davis, and J. Ayliffe. Track initiation and nearest neighbours incorporated into pda. *Journal of Electrical and Electronics Engineers, Australia*, 6(3):191–198, 1986.

[32] D. Comaniciu and P. Meer. Robust analysis of feature spaces: Color image segmentation. *Proc. IEEE Conf. on Comp. Vis. and Pattern Recognition (CVPR)*, pages 750–755, 1997.

[33] D. Comaniciu and P. Meer. Cell image segmentation for diagnostic pathology. *Advanced algorithmic approaches to medical image segmentation: State-of-the-art applications in cardiology, neurology, mammography and pathology*, pages 541–558, 2002.

[34] B. Cramariuc, M. Gabbouj, and J. Astola. Clustering based region growing algorithm for color image segmentation. volume 2, pages 857–860, 1997.

[35] R. Cucchiara, C. Grana, M. Piccardi, and A. Prati. Detecting moving objects, ghosts and shadows in video streams. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 25(10):1337–1342, 2003.

[36] T. Darell, G. Gordon, M. Harville, and J. Woodfill. Integrated person tracking using stereo, color, and pattern detection. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 601–608, 1998.

[37] D. Demandolx and J. Davoust. Multiparameter image cytometry: From confocal micrographs to subcellular fluorograms. *Bioimaging*, 5(3):159–169, 2001.

[38] H. Digabel and C. Lantuejoul. Iterative algorithms. *In Actes du Second Symposium Europeen d'Analyse Quantitative des Microstructures en Sciences des Materiaux Biologie et Medecine*, pages 85–99, 1978.

[39] M. N. Do and M. Vetterli. The finite ridgelet transform for image representation. *IEEE Transactions on Image Processing*, 12(1):16–28, 2003.

[40] D. L. Donoho and I. M. Johnstone. De-noising by soft-thresholding. *IEEE Transactions on Information Theory*, 41(3):613–627, 1995.

[41] D. L. Donoho and I. M. Johnstone. Ideal spatial adaptation via wavelet shrinkage. *Biometrika*, 81(1):425–455, Sept 1994.

[42] A. Doucet, C. Andrieu, and S. Godsill. On sequential monte carlo sampling methods for bayesian filtering. *Statistics and Computing*, 10(3):197–208, 2000.

[43] A. Doucet, N. de Freitas, and N. Gordon. *Sequential Monte Carlo Methods in Practice*. Springer, 2001.

[44] R. O. Duda and P. E. Hart. Use of the hough transformation to detect lines and curves in pictures. *Commun. Assoc. Comput. Mach.*, 15:11–15, 1972.

[45] A. Elgammal, D. Harwood, and L. Davis. Non-parametric model for background subtraction in proc. eur. conf., 2000. In *Europ. Conf. in Computer Vision*, pages 751–767, 2000.

[46] C. Eveland, K. Konolige, and R. C. Bolles. Background modeling for segmentation of video-rate stereo sequences. *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 266–271, 1998.

[47] T. E. Fortmann, Y. Bar-Shalom, and M. Scheffe. Sonar tracking of multiple targets using joint probabilistic data association. *IEEE Journal of Oceanic Engineering*, 8:173–184, 1983.

[48] N. Friedman and S. Russell. Image segmentation in video sequences: A probabilistic approach. *Conference Uncertainty in Artificial Intelligence*, pages 175–181, 1997.

[49] K. Fu and J. Mui. A survey on image segmentation. *Pattern Recognition*, 13:3–16, 1981.

[50] K. Fukunaga and L. Hostetler. The estimation of the gradient of a density function, with applications in pattern recognition. *IEEE Trans. on Info. Theory*, IT-21:32–40, 1975.

[51] P. Fung, W. Lee, and I. King. Randomized generalized hough transform for 2-d grayscale object detection. volume 1, pages 1109–1112, 2002.

[52] J. Gambotto. A new approachto combining region growing and edge detection. *Pattern Recognition Letters*, 14:869 875, 1993.

[53] D. Geman and B. Jedynak. An active testing model for tracking roads in satellite images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(1):1–13, 1996.

[54] J.M. Geusebroek, A.W.M. Smeulders, and F. Cornelissen. Segmentation of cell clusters by nearest neighbour graphs. In *Proceedings of the third annual conference of the Advanced School for Computing and Imaging*, pages 248–252, 1997.

[55] T. Gevers and A.W.M. Smeulders. Combining region splitting and edge detection through guided delaunay image subdivision. In *Proc. of IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 1021–1026, 1997.

[56] C. Glasbey. An analysis of histogram-based thresholding algorithm. *Graphical Models and Image Processing*, 55(6):532–537, 1993.

[57] R. C. Gonzalez and R. E. Woods. *Digital Image Processing*. Prentice Hall, Upper Saddle River, NJ, 2002.

[58] N. J. Gordon, D. J. Salmond, and A. F. M. Smith. Novel approach to nonlinear/non-gaussian bayesian state estimation. *IEE Proceedings-F*, 140(2):107–113, 1993.

[59] R. Haralick and L. Shapiro. Image segmentation techniques. *Computer Vision, Graphics and Image Processing*, 29:100–132, 1985.

[60] R. Haralick and L. Shapiro. *Computer and Robot Vision*, volume 1. Addison Wesley Inc., Reading, Massachussets, 1992.

[61] R. Haralick and L. Shapiro. *Computer and Robot Vision*, volume 2. Addison Wesley Inc., Reading, Massachussets, 1993.

[62] M. Harville, G. Gordon, and J. Woodfill. Foreground segmentation using adaptive mixture model in color and depth. In *IEEE Workshop Detection and Recognition of Events in Video*, pages 3–11, 2001.

[63] M. Heikkila and M. Pietikainen. A texture-based method for modeling the background and detecting moving objects. *IEEE Tran. on PAMI*, 28(4):657–662, 2006.

[64] P. V. C. Hough. *Method and means for recognizing complex patterns.* U.S. Patent 3069654, 1962.

[65] D. Harwood I. Haritaoglu and L. Davis. w^4: Real-time surveillance of people and their activities. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8):809–830, 2000.

[66] N. Ikonomakis, K.N. Plataniotis, and A.N. Venetsanopoulos. Gray-scale and colour image segmentation via region growing and region merging. *Canadian Journal of Electrical and Computer Engineering*, 23(1):43–47, 1998.

[67] J. Ilow and H. Leung. Self-similar texture modeling using farima processes with applications to satellite images. *IEEE Transactions on Image Processing*, 10(5):792–797, 2001.

[68] S. J. Julier and J. K. Uhlmann. A new extension of the kalman filter to nonlinear systems. *in Proceedings of AeroSense: The 11th International Symposium on Aerospace / Defence Sensing, Simulation and Controls*, II, 1997.

[69] P. KaewTraKulPong and R. Bowden. An improved adaptive background mixture model for real-time tracking with shadow detection. In *European Workshop Advanced Video Based Surveillance Systems*, 2001.

[70] Y. Kanai. Image segmentation using intensity and color information. In *Proceedings of The International Society for Optical Engineering (SPIE) - Visual Communications and Image Processing*, pages 709–720, 1998.

[71] T. Kanungo, D. Mount, N. Netanyahu, C. Piatko, R. Silverman, and A. Wu. An efficient k-means clustering algorithm: Analysis and implementation. *IEEE Tran. on Pattern Analysis and Machine Intelligence*, 24(7):881–892, 2002.

[72] K. Karmann and A. V. Brandt. Moving object recognition using an adaptive background memory. In *Time-Varing Image Processing Moving Object Recognition*, pages 289–296, 1990.

[73] J. Kato, T. Watanabe, S. Joga, J. Rittscher, and A. Blake. An hmm-based segmentation method for traffic monitoring movies. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(9):1291–1296, 2002.

[74] J. Kittler and J. Illingworth. Minimum error thresholding. *Pattern Recognition*, 19(1):41–47, 1986.

[75] D. Koller, J. Weber, T. Huang, J. Malik, G. Ogasawara, B. Rao, and S. Russel. Towards robust automatic traffic scene analysis in real-time. In *Proceedings of Intl Conference on Pattern Recognition (ICPR)*, pages 126–131, 1994.

[76] H. W. Kuhn. The hungarian method for the assignment problem. *Naval Research Logistics Quarterly*, 2:83–97, 1955.

[77] C. Lacoste, X. Descombes, J. Zerubia, and N. Baghdadi. A bayesian geometric model for line network extraction. In *IEEE-ICASSP*, volume III, pages 565–568, 2004.

[78] C. Lantuejoul. *PhD thesis: La squelettisation et son application aux mesures topologiques des mosaiques polycristallines*. Ecole des Mines, Paris, 1978.

[79] V. F. Leavers. The dynamic generalized hough transform. *CVIP: Image understanding*, 56(1):381–398, 1992.

[80] D. S. Lee. Effective gaussian mixture learning for video background subtraction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(5):809 830, 2005.

[81] S. Lefevre, C. Dixon, C. Jeusse, and N. Vincent. A local approach for fast line detection. In *IEEE Conference on DSP*, volume 1, pages 1109–1112, 2002.

[82] L. Li, W. M. Huang, I. Y. H. Gu, and Q. Tian. Foreground object detection in changing background based on color co-occurrence statistics. In *IEEE Workshop Applications of Computer Vision*, page 269 274, 2002.

[83] L. Li and M. Leung. Integrating intensity and texture differences for robust change detection. *IEEE Transactions on Image Processing*, 11(2):105–112, 2002.

[84] F. Lin, K. Cordes, L. Li, L. Hood, W. G. Couser, S. J. Shankland, and P. Igarashi. Hematopoietic stem cells contribute to the regeneration of renal tubules after renal ischemia-reperfusion injury in mice. *Journal of the American Society of Nephrology*, 14:1188–1199, 2003.

[85] S. Lloyd. Least squares quantization in pcm. *IEEE Transactions on Information Theory*, 28(2):129–137, 1982.

[86] W.Y. Ma and B.S. Manjunath. Edge flow: A framework of boundary detection and image segmentation. In *Proc. of IEEE Int. Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 744–749, 1997.

[87] J. MacQueen. Some methods for classification and analysis of multivariate observations. In *Proceedings of 5-th Berkeley Symposium on Mathematical Statistics and Probability*, volume 1, pages 281–297, 1967.

[88] S. Mallat. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 11(1):674–693, 1989.

[89] S. Mallat and Z. Zhang. Matching pursuits with time-frequency dictionaries. *IEEE Transactions on Signal Processing*, 41(12):3397–3415, 1993.

[90] T. Markiewicz, S. Osowski, L. Moszczyski, and R. Satat1. Myelogenous leukemia cell image preprocessing for feature generation. In *5th International Workshop on Computational Methods in Electrical Engineering*, pages 70–73, 2003.

[91] N. J. B. McFarlane and C. P. Schofield. Segmentation and tracking of piglets in images. *Machine Vision Applications*, 8:187–193, 1995.

[92] V. Meas-Yedid, F. Cloppet, A. Roumier, A. Alcover, J-C Olivo-Marin, and G. Stamon. Quantitative microscopic image analysis by active contours. In *Vision Interface Annual Conference 2001 - Medical Applications*, 2001.

[93] F. Meyer and S. Beucher. Morphological segmentation. *Journal of Visual Communication and Image Representation*, 1(1):21–45, 1990.

[94] V. Mezaris, I. Kompatsiaris, N.V. Boulgouris, and M.G. Strintzis. Real-time compressed-domain spatiotemporal segmentation and ontologies for video indexing and retrieval. *IEEE Transactions on Circuits and Systems for Video Technology*, 14(5):606 – 621, 2004.

[95] K. G. Murthy. *Network Programming*. Prentice Hall, Englewood Cliffs, NJ., 1992.

[96] G. Nistor, M. Totoiu, N. Haque, M. Carpenter, and H. Keirstead. Human embryonic stem cells differentiate into oligodendrocytes in high purity and myelinate after spinal cord transplantation. *GLIA*, 49(3):385–396, 2004.

[97] N. Otsu. A threshold selection method from gray-level histograms. *IEEE Tran. on Systems, Man, and Cybernetics*, 9:62 66, 1979.

[98] N. Pal and S. Pal. A review on image segmentation techniques. *Pattern Recognition*, 26(4):1277 1294, 1993.

[99] L. Y. Pao. Multisensor multitarget mixture reduction algorithms for target tracking. *AIAA Journal of Guidance, Control and Dynamics*, 17:1205–1211, 1994.

[100] N. Paragios and V. Ramesh. A mrf-based approach for real-time subway monitoring. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1034–1040, 2001.

[101] T. Pavlidis and Y. Liow. Integrating region growing and edge detection. *IEEE Tran. on PAMI*, 12(3):225 233, 1990.

[102] A. Perez and T. Pavlidis. An iterative thresholding algorithm for image segmentation. *IEEE Tran. on PAMI*, 9(6):742 775, 1987.

[103] N. Ray and S.T. Acton. Motion gradient vector flow: An external force for tracking rolling leukocytes with shape and size constrained active contours. *IEEE Transactions on Medical Imaging*, 23(12):1466–1478, 2004.

[104] J. Rehg, M. Loughlin, and K. Waters. Vision for a smart kiosk. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 690–696, 1997.

[105] D. Reid. An algorithm for tracking multiple targets. *IEEE Transactions on Automation and Control*, 24(6):84–90, 1979.

[106] C. Ridder, O. Munkelt, and H. Kirchner. Adaptive background estimation and foreground detection using kalman-filtering. In *Proceedings of International Conference on recent Advances in Mechatronics, (ICRAM)*, pages 193–199, 1995.

[107] J.H. Rife and S.M. Rock. Design and validation of a robotic control law for observation of deep-ocean jellyfish. *IEEE Transactions on Robotics*, 22(2):282–291, 2006.

[108] A. Rosenfeld. *Picture Processing by Computer*. Academic Press, New York, NY, 1969.

[109] A. Rosenfeld and A. Kak. *Digital Picture Processing*, volume 2. Academic Press, New York, NY, 1982.

[110] D. J. Salmond. Mixture reduction algorithms for target tracking in clutter. *Signal and Data Processing of Small Targets, SPIE*, 1305:434–445, 1990.

[111] V. Santibaez, R. Kelly, and M.A. Llama. Global asymptotic stability of a tracking sectorial fuzzy controller for robot manipulators. *IEEE Tran. on SMC Part B: Cybernetics*, 34(1):710–718, 2004.

[112] J. Serra. *Image Analysis and Mathematical Morphology*. Academic Press, New York, NY, 1982.

[113] M. Sezgin and B. Sankur. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, 13(1):146–168, 2004.

[114] Y. Bar Shalom and T. E. Fortmann. *Tracking and Data Association*. Academic-Press, San Diego, CA, 1988.

[115] Y. Bar Shalom, X. R. Li, and T. Kirubarajan. *Estimation with Application to Tracking and Navigation*. Wiley, New York, NY, 2001.

[116] B. Silverman. *Density Estimation for Statistics and Data Analysis*. Chapman and Hall, New York, NY, 1986.

[117] S. K. Sinha, F. Karray, and P. Fieguth. Automated segmentation of underground pipe scanned images. *Proceedings, World Automation Congress*, 3:161–171, 2000.

[118] C. Stauffer and W.E.L. Grimson. Adaptive background mixture models for real-time tracking. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 246–252, 1999.

[119] A. Stuart, K. Ord, and S. Arnold. *Kendall's Advanced Theory of Statistics*. Arnold, a member of the Hodder Headline Group, London, 1999.

[120] H. Tao and T.S. Huang. Color image egde detection using cluster analyis. In *Proc. of 1997 Int. Conf. on Image Processing (ICIP)*, volume I, pages 834–836, 1997.

[121] C. Taswell. The what, how, and why of wavelet shrinkage denoising. *IEEE Journal Computing in Science and Engineering*, 2(3):12–17, 2000.

[122] A. Tremeau and N. Borel. A region growing and merging algorithm to color segmentation. *Pattern Recognition*, 30(7):1191–1204, 1997.

[123] G. Turin. An introduction to matched filters. *IRE Transactions on Information Theory*, 6(3):311– 329, 1960.

[124] L. Vincent and P. Soille. A new approachto combining region growing and edge detectionwatersheds in digital spaces: an efficient algorithm based on immersion simulations. *IEEE Trans. on PAMI*, 13(6):583–598, 1991.

[125] I. L. Weissman, D. J. Anderson, and F. Gage. Stem and progenitor cells: Origins, phenotypes, lineage commitments, and transdifferentiations. *Annual Review of Cell and Developmental Biology*, 17:387–403, 2001.

[126] C.R. Wren, A. Azarbayejani, T. Darrell, and A.P. Pentland. Pfinder: Real-time tracking of the human body. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 19(7):780–785, 1997.

[127] H. S. Wu, J. Barba, and J. Gil. Region growing segmentation of textured cell images. *IEE Electronics Letters*, 32(12):1084–1085, 1996.

[128] K. Wu, D. Gauthier, and M. Levine. Live cell image segmentation. *IEEE Transactions on Biomedical Engineering*, 42(1):1–12, 1995.

[129] Y. Xiaohan, J. Yla-Jaaski, O. Huttunen, T. Vehkomaki, and O. Sipild T. Katila. Image segmentation combining region growing and edge detection. In *International Conference on Pattern Recognition (ICPR)*, volume C, page 481 484, 1992.

[130] L. Xu and E. Oja. Randomized hough transform. *CVIP Image underestanding*, 57(2):131–154, 1993.

[131] Q. Zang and R. Klette. robust background subtraction and maintenance. In *International Conference Pattern Recognition (ICPR)*, pages 90–93, 2004.

[132] Z. Zivkovic. Improved adaptive gaussian mixture model for background subtraction. In *International Conference on Pattern Recognition (ICPR)*, pages 28–31, 2004.

[133] D. Zugaj and V. Lattuati. A new approach of color image segmentation based on fusing region and edge segmentations outputs. *Pattern Recognition*, 31(2):105–113, 1998.