# Analyse du potentiel migratoire de cellules cancéreuses par prétraitement, segmentation et classification d'images

Tahir Qasim Syed

## To cite this version:

Tahir Qasim Syed. Analyse du potentiel migratoire de cellules cancéreuses par prétraitement, segmentation et classification d'images. Traitement du signal et de l'image [eess.SP]. Université d'Evry-Val d'Essonne, 2011. Français. NNT : . tel-00847466

HAL Id: tel-00847466
https://theses.hal.science/tel-00847466v1
Submitted on 23 Jul 2013

UNIVERSITÉ D'EVRY VAL D' ESSONNE
Laboratoire d'Informatique, Biologie Intégrative et Systèmes Complexes

Thesis submitted for the degree of Doctor of Philosophy (PhD)
Université d'Evry-Val d'Essonne

# Analysis of the migratory potential of cancerous cells by image preprocessing, segmentation and classification

SYED Tahir Qasim
Defended on : 13/12/2011

JURY

N. Vincent : Professor Université Paris 5, Reviewer
O. Lezoray : Professor, Université de Caen, Reviewer
E. Petit : Professor Université de Paris 12 Créteil, Examiner
G. Barlovatz-Meimon : Professor, Université de Paris 12 Créteil, Examiner
J. Triboulet : Assistant Professor Université de Nîmes, Examiner
V. Vigneron : Assistant Professor, Université d'Evry, Co-Supervisor
C. Montagne : Assistant Professor, Université d'Evry, Co-Supervisor
S. Lelandais-Bonadè : Professor, Université d'Evry, Supervisor

# Abstract

This thesis is part of a broader research project which aims to analyze the potential migration of cancer cells. As part of this doctorate, we are interested in the use of image processing to count and classify cells present in an image acquired using a microscope. The partner biologists of this project study the influence of the environment on the migratory behavior of cancer cells from cell cultures grown on different cancer cell lines. The processing of biological images has so far resulted in a significant number of publications, but in the case discussed here, since the protocol for the acquisition of images acquired was not fixed, the challenge was to propose a chain of adaptive processing that does not constrain the biologists in their research. Four steps are detailed in this paper. The first concerns the definition of pre-processing steps to homogenize the conditions of acquisition. The choice to use the image of standard deviations rather than the brightness is one of the results of this first part. The second step is to count the number of cells present in the image. An original filter, the so-called “halo” filter, that reinforces the centre of the cells in order to facilitate counting, has been proposed. A statistical validation step of the centres affords more reliability to the result. The stage of image segmentation, undoubtedly the most difficult, constitutes the third part of this work. This is a matter of extracting images each containing a single cell. The choice of segmentation algorithm was that of the “watershed”, but it was necessary to adapt this algorithm to the context of images included in this study. The proposal to use a map of probabilities as input yielded a segmentation closer to the edges of cells. As against this method leads to an over-segmentation must be reduced in order to move towards the goal: “one region = one cell”. For this algorithm the concept of using a cumulative hierarchy based on mathematical morphology has been developed. It allows the aggregation of adjacent regions by working on a tree representation of these regions and their associated level. A comparison of the results obtained by this method with those proposed by other approaches to limit over-segmentation has allowed us to prove the effectiveness of the proposed approach. The final step of this work consists in the classification of cells. Three classes were identified:

spread cells (mesenchymal migration), “blebbing” round cells (amoeboid migration) and “smooth” round cells (intermediate stage of the migration modes). On each imagette obtained at the end of the segmentation step, intensity, morphological and textural features were calculated. An initial analysis of these features has allowed us to develop a classification strategy, namely to first separate the round cells from spread cells, and then separate the “smooth” and “blebbing” round cells. For this we divide the parameters into two sets that will be used successively in two the stages of classification. Several classification algorithms were tested, to retain in the end, the use of two neural networks to obtain over 80% of good classification between long cells and round cells, and nearly 90% of good classification between “smooth” and “blebbing” round cells.

# Résumé

Ce travail de thèse s'insère dans un projet de recherche plus global dont l'objectif est d'analyser le potentiel migratoire de cellules cancéreuses. Dans le cadre de ce doctorat, on s'intéresse à l'utilisation du traitement des images pour dénombrer et classifier les cellules présentes dans une image acquise via un microscope. Les partenaires biologistes de ce projet étudient l'influence de l'environnement sur le comportement migratoire de cellules cancéreuses à partir de cultures cellulaires pratiquées sur différentes lignées de cellules cancéreuses. Le traitement d'images biologiques a déjà donné lieu à un nombre important de publications mais, dans le cas abordé ici et dans la mesure où le protocole d'acquisition des images acquises n'était pas figé, le défi a été de proposer une chaîne de traitements adaptatifs ne contraignant pas les biologistes dans leurs travaux de recherche. Quatre étapes sont détaillées dans ce mémoire. La première porte sur la définition des prétraitements permettant d'homogénéiser les conditions d'acquisition. Le choix d'exploiter l'image des écarts-type plutôt que la luminosité est un des résultats issus de cette première partie. La deuxième étape consiste à compter le nombre de cellules présentent dans l'image. Un filtre original, nommé filtre «halo», permettant de renforcer le centre des cellules afin d'en faciliter leur comptage, a été proposé. Une étape de validation statistique de ces centres permet de fiabiliser le résultat obtenu. L'étape de segmentation des images, sans conteste la plus difficile, constitue la troisième partie de ce travail. Il s'agit ici d'extraire des «vignettes», contenant une seule cellule. Le choix de l'algorithme de segmentation a été celui de la «Ligne de Partage des Eaux», mais il a fallu adapter cet algorithme au contexte des images faisant l'objet de cette étude. La proposition d'utiliser une carte de probabilités comme données d'entrée a permis d'obtenir une segmentation au plus près des bords des cellules. Par contre cette méthode entraine une sur-segmentation qu'il faut réduire afin de tendre vers l'objectif : «une région = une cellule». Pour cela un algorithme utilisant un concept de hiérarchie cumulative basée morphologie mathématique a été développé. Il permet d'agréger des régions voisines en travaillant sur une représentation arborescente de ces régions et de leur niveau associé. La comparaison des résultats

obtenus par cette méthode à ceux proposés par d'autres approches permettant de limiter la sur-segmentation a permis de prouver l'efficacité de l'approche proposée. L'étape ultime de ce travail consiste dans la classification des cellules. Trois classes ont été définies : cellules allongées (migration mésenchymateuse), cellules rondes «blebbantes» (migration amiboïde) et cellules rondes «lisses» (stade intermédiaire du mode de migration). Sur chaque vignette obtenue à la fin de l'étape de segmentation, des caractéristiques de luminosité, morphologiques et texturales ont été calculées. Une première analyse de ces caractéristiques a permis d'élaborer une stratégie de classification, à savoir séparer dans un premier temps les cellules rondes des cellules allongées, puis séparer les cellules rondes «lisses» des «blebbantes». Pour cela on divise les paramètres en deux jeux qui vont être utilisés successivement dans ces deux étapes de classification. Plusieurs algorithmes de classification ont été testés pour retenir, au final, l'utilisation de deux réseaux de neurones permettant d'obtenir plus de 80% de bonne classification entre cellules longues et cellules rondes, et près de 90% de bonne classification entre cellules rondes «lisses» et «blebbantes».

# Contents

# List of Figures

# List of Tables

# Introduction

Biology is profiting from mathematics and engineering especially through the automation of labour-intensive tasks. A successful example can be found, for instance, in cell migration analysis and drug testing areas. The present work focuses on the different digital image processing technologies available that open the possibility to monitor and to characterize the migratory behavior of cancer cells. Cancer cell observations have been extensively used for many years in a wide range of applications, including cell migration analysis and drug testing. Nowadays, computer assisted-microscopy allows the handling of considerably large amounts of image data acquired during experiments lasting over several hours or indeed several days. The combination of time-lapse microscopy with adapted image analysis methods constitutes an efficient tool for the screening of cell behavior in general, and cell motility and invasiveness in particular.

Cells are either studied as a part of the tissue structure or implanted on an artificial substrate. Our area of concern is an *in vitro* study i.e. the cells have been isolated from their biological source for the purpose of more detailed, controllable and convenient study. In the experiments that form the source for our data, living cells are considered.As processors of image information, our work occupies a context of research and experimental investigation. It focuses therefore on achieving an understanding of which image processing methods are best adapted for the purpose of cell sorting in the given biological context, rather than maximizing processing throughput for example, although these form part of the secondary considerations.

This work was motivated by a question concerning migration of cancerous cells. These cells are clones of the cell that initiated the cancer, having acquired certain characteristics allowing it to divide indefinitely and be able to *metastase* i.e. to proliferate and migrate. Cancer cell migration is itself of two types: *mesenchymatic* and *amoeboid.* Amoeboid migration is fast and is usually responsible for metastasis and development of secondary tumours, while mesanchematic migration results in proliferation within the same tumour. Changes in migratory behaviour are through experimental observation associated with phenotypes or morphologies of the metastasic cells, namely *blebbing*, pertaining to amoeboid migration, *splayed*, pertaining to mesenchymatic migration, and the intermediate *smooth round* phenotype that could range between being perfectly round to slightly elongated.

The experimental goal is to determine how many and which cells are in each of the three phases of the metastasic process. This general objective spurs us toward more concrete sub-objectives of being able to recognize parts of the image as cells (*cell counting*), to separate cells from the image background and from other cells (*cell segmentation*) in order to study their characteristics that represent the 3 phenotypes, finally to recognize the cells into differentiable categories (*cell classification*) according to their metastastic stage. This process determines the numbers and thus proportions of each of the 3 types of cells over an entire image, by first translating the overall problem into sub-problems concerning individual cells, and then re-combining those individual analyses into the global picture of the process of metastasis.

Various authors have developed a panoply of methods for each of the aforementioned sub-tasks in their application contexts. Counting of cells has been described using blob detection, template matching and learning methods to distinguish pixel patches as cells for instance. Cell image segmentation is a classical area of interest with very varied methods, ranging from pixel-classification approaches and thresholding, to edge-detection methods comprising image-feature representation models such as active contours, to region-based approaches such as active regions and the watershed transform. Classification of cells is quite often the end goal of many cytology applications, employing adapted cellular characteristics. Applications similar to ours that federate an array of different methods are also found in the literature. We shall visit the works of these authors in the following chapters. But what makes our problem different is the nature of the metastasic cells. They exhibit a larger array of shapes and orientation, are harder to discern from the image background and tend to adhere into cell-agglomerates because of the ongoing cellular processes, unlike the cells in blood smears on which the majority of the studies in the literature have focused.

The special biological context and image acquisition conditions demanded that bespoke methods be developed, adapted to the demanding application context. Hence, following a number of pre-processing measures adapted to enable better exploitation of the image data, a template-matching “halo” filter has been developed to accentuate and thereby detect the cells on an image, and a log-likelihood test has been put into place that measures the degree of the match to validate the detection, permitting an efficient and precise counting of the total number of cells.

Image segmentation offered the greatest challenge to a correct resolution of the problem since determining the cell's metastasic type requires precise morphological information, and a specialized method of the watershed transformed that we call "cumulative hierarchy" has been developed that outperforms the usual approaches taken by authors by a significant degree for the given image data. Finally, a set of cellular characteristics has been conceptualized and used to classify the cells into each metastasic phenotype.

The details of these methods are elucidated along five chapters. The thesis is organized by processing stage, i.e. pre-processing, cell detection, image segmentation and classification, and each chapter treats both the related literature as well as our methods and their benchmarking with respect to some alternatives from the literature. In the following, we briefly summarize the contents of chapters.

*Chapter* 1: This chapter serves as a general review of tools and methods employed by cellular biologists and the manner in which image processing technologies help them in various areas of the different stages of their work. This chapter therefore lays the foundation on which the work we present in the following chapters could be established. Once we have shown *why*, and equally importantly, *how* these images have been obtained, we funnel toward our specific experimental context and objectives.

*Chapter* 2: The goal in this chapter is to develop an automatic cell detection technique that could supplant human intervention while attaining comparable accuracy. This is a particularly useful step for biologists studying the evolution of cancer under varying environmental conditions since it saves them arduous work. However, the various difficulties the data pose are first overcome through tailored pre-processing, which preceded the discussion on cell detection. The chapter concludes with a validation scheme for the cells detected by our filtering approach, and a comparison with a known method. The output of the chapter are cell locations and counts, as well as a binary image that distinguishes image pixels belonging to cellular agglomerates from those in the image background.

*Chapter* 3: This chapter takes us through the segmentation mechanism we have devised to separate cells among them. Thus the data we exploit from the previous chapter comprises: the original grey level image, the binarised image of agglomerate connected components, and the location of the cell centres detected and validated for the cells in the image. The chapter offers a review of image segmentation algorithms in the context of cellular images, and explains our choice of segmentation method. Then it proposes several algorithms for the application of the proposed method, including one original proposal, a sum of fuzzy probabilities map, and then numerically and qualitatively compares them to decide on the one we will ultimately use for segmentation. This evaluation uses a segmentation quality criterion we define in the chapter. This application methodology, although precise in describing cell boundaries suffers from the drawback of fragmenting the image into far more segments than is required. The following chapter aims to redress this problem.

*Chapter* 4: The chapter begins by explaining the problem of over-segmentation and its sources in our data. Then it offers two alternative possibilities to correct it, one involving modifying the image function before or during the segmentation process using mathematical morphology, and the other that initially allows over-segmentation and then tries to resolve it by combining image fragments according to various rules. Two methods for the former and four for the latter are detailed and implemented. We propose our own algorithm that combines the first two together to produce a flexible segmentation approach that removes the drawbacks of either. An evaluation is performed for the five segmentation refinement algorithms and the most appropriate is retained for the actual segmentation. Segmentation thus performed produces image segments one for each individual cell. This allows the calculation of various classification attributes from these image segments in the following chapter.

*Chapter* 5: This chapter adds the concluding aspect to the work. At this stage we have the number of cells their coordinates on an image from Chapter 2, and their set of connected image pixels that represent a cell as a binary mask as well as the cutout from the original cellular image representing grey level information from Chapter 4. This information is exploited in this chapter to extract discriminatory knowledge about the morphology, grey level and texture of each cell using characteristics that we describe. The most salient characteristics then selected, and passed onto a classifying algorithm in order to decide the metastasic morphology for each cell.

To conclude the document, a chapter of conclusions will resume the principal results obtained within this work and open it up for future perspectives.

# 1

# Situating the problem

Cancer is a major health problem for mankind, and the existing approaches - surgery and radiation - to its treatment have clear limitations, notably early detection and localization [69]. As emphasized by Gibbs in that paper, the past decades have seen a tremendous increase in our knowledge of the molecular mechanisms and patho-physiology of human cancer. Cancer kills patients essentially because of the migratory nature of the cells it effects. Indeed, it is now well established that cell migration plays pivotal roles in cancer cell scattering, tissue invasion and metastasis [150, 23, 112], i.e., processes which are essentially responsible for the dismal prognoses of a majority of cancer patients [80]. The identification of compounds partaking in the migratory process requires adapted *in vitro* and *in vivo* biological models, as well as efficient screening technologies. Concerning the latter, cellular imaging nowadays clearly appears to be an efficient tool for a wide screening of cell behaviour in general, and cell migration in particular. The recent advances and developments in microscopy, cell staining and imaging technologies now allow cell monitoring in increasingly complex environments, which in turn allow the use of more realistic biological models for studying cancer cell migration. Combined with adapted methods of image analysis, this approach is able to provide direct, primary and quantitative information on the effects of various compounds on the migration of cancer cells, and also of other cell actors involved in cancer invasion [126].

## 1.1 Imaging cancer cell migration and associated rare cellular events

In this section we briefly present different levels at which cell migration-related events can be observed, imaged and then analyzed. This description follows general to specific aspects i.e. from an analysis of a global cell population to a focus on a single cell, via intermediary stages centered on individual cell locomotion and related morphological characteristics.

### 1.1.1 Cancer cells as non-static populations colonizing their neighbourhoods

A first level of investigation concerns the analysis of the migratory behavior of a population of cells taken as a whole. The global migration property [116] of a cell population usually refers to its ability to colonize its neighborhood. This ability is generally evaluated as the distance covered by the migration front from the initial site after a predefined period of culture, or the net increase in the total area covered by all the cells. This colonization ability is clearly affected by migration and growth. At this first level of investigation, single cell locomotion is thus not considered, in contrast to the second level described below.

### 1.1.2 Cancer cells as individual moving objects

A second level of analysis focuses on the tracking of individual cells, aiming to reconstruct their trajectories from a set of successive positions. This task encounters a series of difficulties due to phenomena such as cell division, path-crossing and clustering, in addition to the fact that a number of cells may enter and/or exit the observed microscope field.

While being more complex, the analysis of individual cell trajectories has a number of advantages [48]. Firstly, it enables cell migration to be distinguished from cell growth. In addition, by analyzing individual cell migration behavior, it is possible to identify subpopulations of cells presenting different migratory characteristics.Finally, establishing cell trajectories simplifies the detection of preferential directions followed by moving cells, e.g., in response to a chemical agent having chemo-attractive or repulsive properties (one of such is the $PAI - 1$ molecule we shall visit later in the chapter).

### 1.1.3 Characterizing phenotypic and morphologic features of cancer cells

During migration, cancer cells exhibit a variety of morphologic changes. These morphologic changes are characteristic of the various migration modes that the cells could adopt, with possible transitions between them 1.15, [48, 63]. In the case of an **amoeboid** migration mode, amoeboid-like migrating cells use a fast ’crawling’ type of movement, requiring rapid cycles of morphologic expansion and contraction on the part of the cell body. In contrast, the **mesenchymal** mode of cell migration presents a succession of multiple stages involving cell polarization, protrusion extension, cell elongation and contraction processes to allow for cell translocation. In the case of collective migration, cells maintain their cell-cell junctions and move as connected multicellular sheets, aggregates or clusters, in which a promigratory subset of cells at the leading edge can be identified [142], [186]. Consequently, comparative analysis between the cellular ability to migrate and cellular morphologic appearance may provide interesting information on the cell migration process itself, as well as on the influence of the cell environment on this process, in addition to the possible anti-migratory effects of a given compound [186, 108].

### 1.1.4 Combination of the investigation levels

Of course, the combination of different investigation levels appears interesting in order to better characterize cancer cell migration processes and their response to pro-migratory and anti-migratory chemicals. For example, Rosello et al. [172] encourage the use of multi-assay strategies combining data obtained at either the cell population or the individual cell level. On the side of imaging techniques, mixing two- and three-dimensional environments for cell migration observations [48] is recommended.

## 1.2 Microscopy and image acquisition technologies for culture visualization

Given the fundamental importance of cell locomotion, a number of *in vitro* methodologies have been developed to characterize this phenomenon more easily and to allow the study of the effects of endogenous or exogenous molecules on cell migration. *In vitro* tests are generally used to provide a range of initial information because *in vivo* tests are both more difficult and time- and money-consuming to perform, factors that limit the number of tests that can be run at any one time. In addition to this, quantification in *in vivo* tests is also generally more difficult. This is the reason why *in vivo* tests are generally used as the ultimate stage to confirm information provided by in vitro tests.

Two-dimensional *in vitro* models are used to analyze the motility of a cell population in a $2D$-environment, i.e., cells cultured on the surface of culture plates or in wells in fluid environments. Even though increasing evidence suggests that migration across planar substrates is very different from *in vivo* cell behaviour [13], $2D$ cell migration models continue to be in frequent usage for convenience's sake.

### 1.2.1 Culture visualization

One of the main challenges in biology is the ability to observe essentially transparent cell or tissue materials. Solving this practical issue requires adapted methods which vary depending on whether the analyzed (transparent) materials are fixed or not.

In the case of *fixed materials*, standard staining techniques can be easily used to enhance the optical density of the region of interest. In the context of cell migration analysis, this requires the stopping of the experiments after a given period of time (end-point analysis), the fixation and then the staining of the cells. Depending on the purpose and the target of the analysis, different staining techniques are available. Standard cell staining methods (e.g., with cristal violet or toluidin blue) can be used if the aim is simply to identify the cell locations (or the cell number) on 2D transparent supports (*in vitro* models). Fig. 1.1 shows an example of such an image, stained with cristal violet. Digitized images of the stained cultures can be easily acquired using standard light microscopy and then submitted to image analysis for quantification see later section 1.5.3.B on our image acquisition).

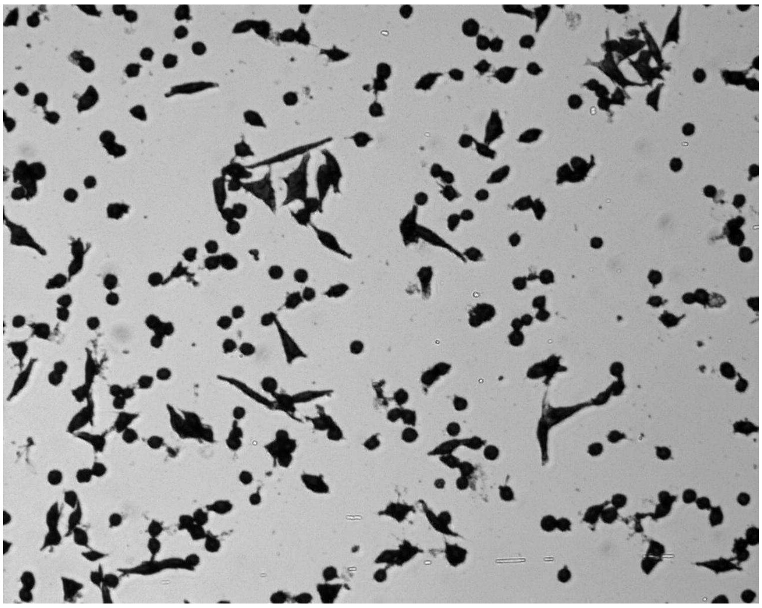


**Figure 1.1:** *Migrating cells fixed and stained with gentiane/cristal violet under light microscopy.*

In contrast, the monitoring of unstained living specimens requires other techniques. This is why microscopists have developed several optical tricks to exploit refraction differences that may exist between living material and its surrounding environment. Techniques such as phase-contrast microscopy and differentiated interference contrast enable contrasted images to be obtained from transparent specimens [237] (staining is a difficult and time consuming procedure which sometimes, but not always, destroys or alters the specimen.). These techniques make possible the time-lapse monitoring of marker-free/unstained living cells. This approach usually consists of automatically recording frame sequences of living cell cultures through relatively inexpensive microscopes equipped with video acquisition systems, such as the one we will take as the example throughout this thesis (Fig. 1.19). All the *in vitro* migration based on cells cultured in transparent 2D environments can be easily monitored with this approach [193].

Fluorescence-staining techniques have been adapted to living cells [193]. More particularly, genetically encoded fluorophores, such as the green fluorescent protein (GFP) and its color-shifted genetic derivatives, can be used to tag biomolecules [195], making their tracking in living systems easier. This enables monitoring of cellular processes by means of live-cell imaging experiments based on fluorescence microscopy [193]. Refer to Fig. 1.2 for an image of metastasic cells marked by the *Papanicolaou* stain, showing distinctly the membrane, cytoplasm and nucleus.

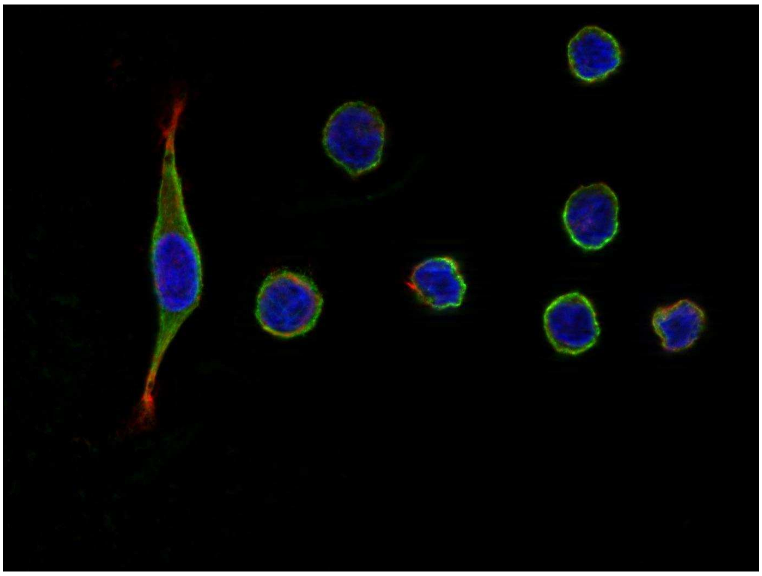


**Figure 1.2:** *Fluorescent-marked migrating cells.*

If the aim is simply to mark cells in order to facilitate their tracking through less transparent or opaque substrates (such as tissue), simpler approaches can be used which do not require fluorescent protein fusion products. Of these, fluorescent vital dyes (e.g., the $1,1'-dioctadecyl-3,3,3',3'-tetramethylindocarbocyanineperchlorate$) are able to bind to cellular membranes of living cells and thus clearly delineate the entire cell morphology [195].

By enhancing only the object/target of interest, fluorescence microscopy has numerous advantages, such as allowing trivial image processing techniques (like classical image threshold, see section 1.3.2).

### 1.2.2 Characterisation of imaging techniques

Before going on to quantitative image analysis methods in the following section, we want to conclude the present section by highlighting a number of key points. In addition to the different investigation levels described in section 1.1 at which cell events can be imaged and analyzed, the different imaging techniques can be characterized according to the following key points:

**The contrast method**: As mentioned above, cells are mainly transparent and thus require systems to generate contrast. Two kinds of approaches can thus be distinguished: on one hand, optically enhanced microscopy methods and, on the other hand, stained cell imaging (by using fluorescent or other cell markers).

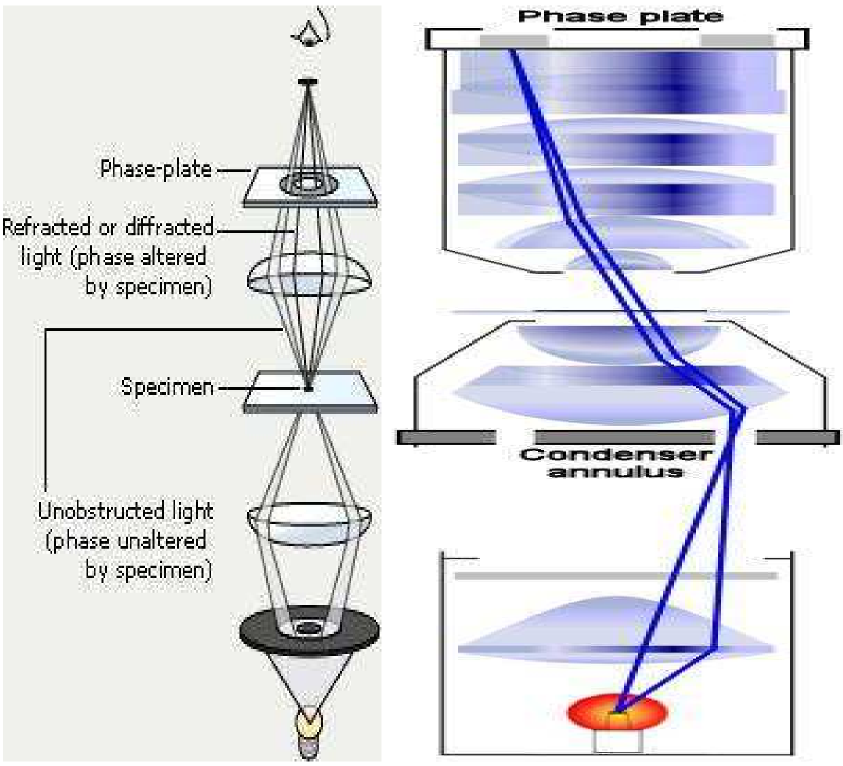
**Time monitoring**: Another consideration in imaging is the way time is taken into account. End-point cell analysis consists of analyzing samples after a period of time. This allows the samples to be fixed (in their current states) and a contrasting compound to be used to reveal or stain the targets of interest. Measures with time can also be achieved by stopping a number of cell culture replicates (carried out under the exact same conditions) after different time periods. In contrast, in a time-lapse analysis, the living cells are observed uninterruptedly over time, enabling continuous processes to be monitored.

**The acquisition depth**: Single images ($2D$) can be acquired, such as in the case of end-point applications, or image sequences for time-lapse ($2D+T$), or even sequences of image stacks ($3D$). These two latter cases are used in monitoring cell processes occurring over time in 2D and 3D environments, respectively.

### 1.2.3 Phase-contrast microscopy

Phase contrast is a widely used optical microscopy illumination technique that shows differences in refractive index i.e. small phase shifts in the light passing through a transparent specimen as amplitude or contrast changes in the image [44]. It was developed by the Dutch physicist F. Zernike in the 1930s [235, 236]. The phase contrast microscope is a vital instrument in biological and medical research. When dealing with transparent and colorless components in a cell, dyeing is an alternative but at the same time stops all processes in it. The phase contrast microscope has made it possible to study living cells, and cell division is an example of a process that has been examined in detail with it. The phase contrast microscope was awarded with the Nobel Prize in Physics, 1953[237].

The phase contrast microscope uses the fact that the light passing trough a transparent part of the specimen travels slower and, due to this, is shifted compared to the uninfluenced light. However, the change in phase can be increased to half a wavelength by a transparent phase-plate in the microscope and thereby causing a difference in brightness. Changes in amplitude give rise to familiar absorption of light, which is wavelength dependent and gives rise to colours. The human eye measures only the energy of light arriving on the retina, so changes in phase are not easily observed, yet often these changes in phase carry a large amount of information. The nucleus in a cell for example will show up dark against the surrounding cytoplasm. Contrast is excellent; however the technique cannot be used with thick objects that attenuate the light or possess opacity. Frequently, a halo is formed even around small objects, which obscures detail, a feature of which our images are good examples (Fig. 1.4).



(a) (b)

**Figure 1.3:** *A practical implementation of phase-contrast illumination consists of a phase ring (located in a conjugated aperture plane somewhere behind the front lens element of the objective) and a matching annular ring, which is located in the primary aperture plane (location of the condenser's aperture).*

The system consists of a circular annulus in the condenser, which produces a cone of light (Fig. 1.3 a.). This cone is superimposed on a similar sized ring within the phase-objective. Every objective has a different size ring, so for every objective another condenser setting has to be chosen. The ring in the objective has special optical properties: it first of all reduces the direct light in intensity, but more importantly, it creates an artificial phase difference of about a quarter wavelength. As the physical properties of this direct light have changed, interference with the diffracted light occurs, resulting in the phase contrast image. Fig. 1.3 b. shows a cross section of the illuminator, condenser and objective. Two selected light rays (indicated by

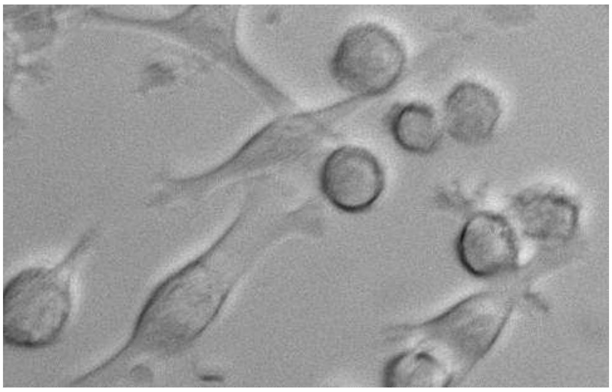


**Figure 1.4:** *Example of image details being obstructed by a halo of light formed around objects in phase-contrast microscopy.*

blue lines), which are emitted from one point inside the lamp's filament, get focused by the field lens exactly inside the opening of the condenser annular ring.

In summary, with the development of highly sensitive and high-throughput imaging instruments, microscopy has become the major tool to study cellular distributions and interactions. This chapter presented in its first four sections the ongoing biotechnology body of research that promises an understanding of the mechanisms, and in the following two sections a brief overview of the state-of-the-art of how the corpus of data produced by these technologies is exploited by researchers in the field of image-processing to further understand, in a symbiotic manner to the biologists that study them, these aforementioned cellular mechanisms.

## 1.3 Quantitative cell image analysis

Digital image processing and analysis is able to summarize a large amount of images into a few, hopefully meaningful and essentially numerical descriptors. As detailed below, cell image analysis is usually a chained process beginning with low-level preprocessing, followed by segmentation (i.e., extraction of the candidate objects from the background), the post-processing of the candidate objects, and finally feature extraction that supplies a latter stage of data analysis.

### 1.3.1 Low-level image processing and preprocessing

Specific preprocessing steps are generally needed depending on the type of the acquired images. For instance, optical phase-contrast images are often subject to illumination problems and poor image contrast. As authors have previously suggested [86, 47], a succession of image preprocessing steps are able to remediate these problems. These steps essentially include image background detection, background masking and local grey level histogram equalization. We shall look at those necessitated by the imagery in this work in Chapter 2.

Fluorescence microscopy generates different sources of noise that have to be suppressed. For example, the culture medium or substrate (such as tissue) may have autofluorescence abilities. The resulting noise can be subtracted by estimating the mean background contribution. Finally, a part of the acquired fluorescence comes from out-of-focus planes and should be removed from the image to increase its sharpness.

Some defects originating from phase-contrast microscopy or from the conditions of image acquisition could be reduced by such low-level pre-processing. For example, the intensity non-uniformities mentioned above are treated thus by [218], as Fig. 1.5 illustrates. $A$: An image before background correction and $B$: a spline surface fitted to the image background. $C$: The image after background subtraction. Note that the intensity scale (same for all images) has been set to visually enhance the contrast of the darker parts of the images.

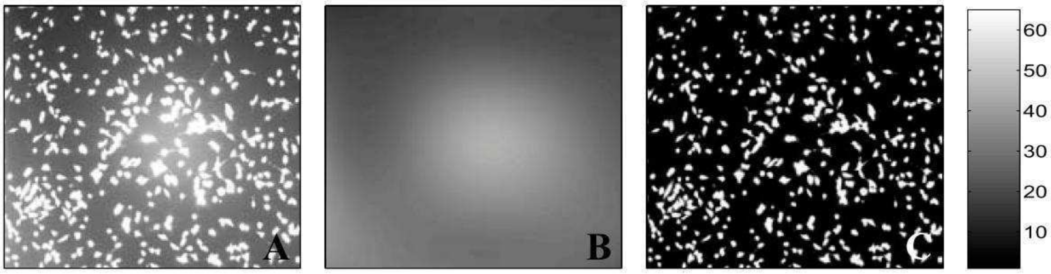


**Figure 1.5:** *Example of the reduction of grey level non-uniformity. Courtesy C. Wahlby [218]*

### 1.3.2 Image segmentation, object detection

Image segmentation consists of the partitioning of the image space into *connected components* belonging to either the objects of interest (e.g. cells) or the background. Very broadly, two families of methods exist to carry out segmentation. Whereas the first is based on the grouping of pixels sharing certain similarities (according to a criterion defined with respect to the image or to the application), the second exploits the borders existing between the objects and the background. The simplest way of exploiting pixel similarities is to use a *grey level threshold* to select the pixels belonging to the objects. This requires the images to present a good level of contrast between the objects and the background, a condition which is not guaranteed in unstained cell imaging. Threshold determination can be based on different techniques relating to various, often statistical, concepts (e.g. area, mean grey level, maximum entropy, clustering, etc.). To efficiently segment cells from the background, authors have used contour detection and watershed transformation, as we shall shortly see in this section. The *watershed transform* is a powerful method of image partitioning but is highly sensitive to the presence of small variations in the images, resulting in image over-segmentation. Different methods were thus developed to circumvent this problem, as we shall see in the Chapters 3 and 4.

Work on the detection and counting of cells in microscopy images is very varied but has mostly been focused on segmentation of cells leading as a byproduct to an automatic count. To begin with, hardware methods [178, 122] exist to identify and quantify sections of cells cultured in suspension. However, being integrated into the material, they are monetarily expensive and require a trained technical specialist.

Several researchers have been developing automated methods for segmenting and counting cells in microscopy images [184, 243, 66, 41]. Anoraganingrum [41] performed edge detection on melanoma cells using median filter and mathematical morphology. Garrido et al. [66] approximated red blood cell locations using a parametric ellipse model and refined its contours using a deformable template. Sheikh et al. [184] proposed a method of identifying the major blood cell types using median and edge enhancing filters and to classify them based upon their morphological features using neural networks. Zimmer et al. [243] suggested tracking of motile cells using a parametric active contour model, along with a comprehensive strategy of working with cellular images. Some approaches are based on machine learning [124, 202, 135, 242]. Long et al. [124] and Zheng et al. [242] proposed methods based on neural network. Markiewicz et al. [135] proposed a method to cell recognition and count using Support Vector Machine. In this kind of approach, the major task is to create the learning set, which is usually done manually by an independent expert for cell type and is time consuming. Another disadvantage is the time spent on training and parameter adjustments. Approaches that use classical segmentation methods, such as threshold, morphological filtering and watershed transformation Korzynska [103] presented a method for automatic counting of neural stem cells growing in cultures which is performed in two steps: a segmentation step where the image is separated in several regions and a counting step where each extracted region is counted as a single cell. Here, as is the case usually, counting and detection are obvious byproducts of image segmentation. Figures 1.6 1.7 show this process from [57].

Blood smear image analysis has been tackled by using conventional image pro-

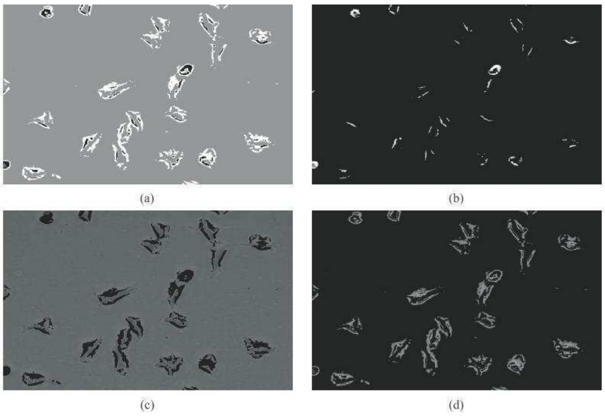


**Figure 1.6:** *Clustering segmentation method from [57], a) clusterized image (*3 *clusters), and one cluster image in b) class* 1*, c) class* 2*, d) class* 3*.*

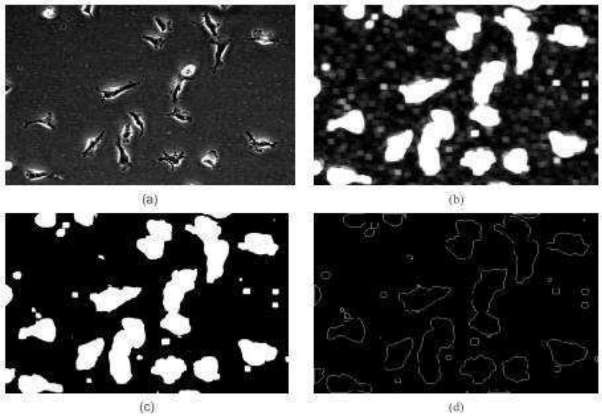


**Figure 1.7:** *a) Histogram-corrected Image , b) image of local variance, c) binary image of variance, d) image of contours.*

cessing techniques like morphology [52], edge detection [187], region growing [199] etc., which all have shown certain degrees of success with respect to the used data, which is arguably ideal for such processing, on account for cells being clearly distinguished from the background and of a non-adhering nature.

Sio et al.[187] addressed the problem of parasitemia estimation using edge detection and splitting of large clumps made up from erythrocytes. The outcome of the approach was shown to be satisfactory for well-stained samples with well-separated erythrocytes. For the same problem, watershed transform [215] were also employed, given that local maxima indicate the centers of convex shapes, i.e. blood components particularly erythrocytes. This concept, however, is only justifiable for images which exhibit a small degree of cell overlap.

Post-processing stages are often needed after image segmentation in order to better identify the objects of interest. These stages aim to separate neighboring objects which remain grouped after segmentation, to merge two parts of the same (over-segmented) object, to fill small holes, to remove small objects, etc. These tasks are usually achieved by means of the so-called “morphological” operators which are usually applied to binary (i.e. segmented) images. While these stages are particularly useful for evaluating certain measurements, such as the object count, they are not necessary for others, such as the measurement of surfaces. Morphologically specialized filters are also used to enhance the characteristics of biological objects, such as actin fibers [118], as for example used by Helmke et al [88] to characterize intermediate filament networks in living cells by thinning image objects to identify their morphological “skeletons”.

High-level image processing such as image segmentation and object detection, coupled with techniques for low-level image processing, are responsible for the bulk of the aid that image processing brings to the table for medical researchers. We shall present in the rest of the thesis an example of such aids that we have been able to furnish in the context of difficult data to the problem presented in section 1.5. For an explanatory example, please examine Figures 1.8 and 1.9, courtesy C. Wahlby [218].

## 1.4 Feature extraction

We review here a number of quantitative features that can be extracted from cell images that are able to provide information on cell migration processes. It must be understood that in its traditional meaning in pattern recognition and in image

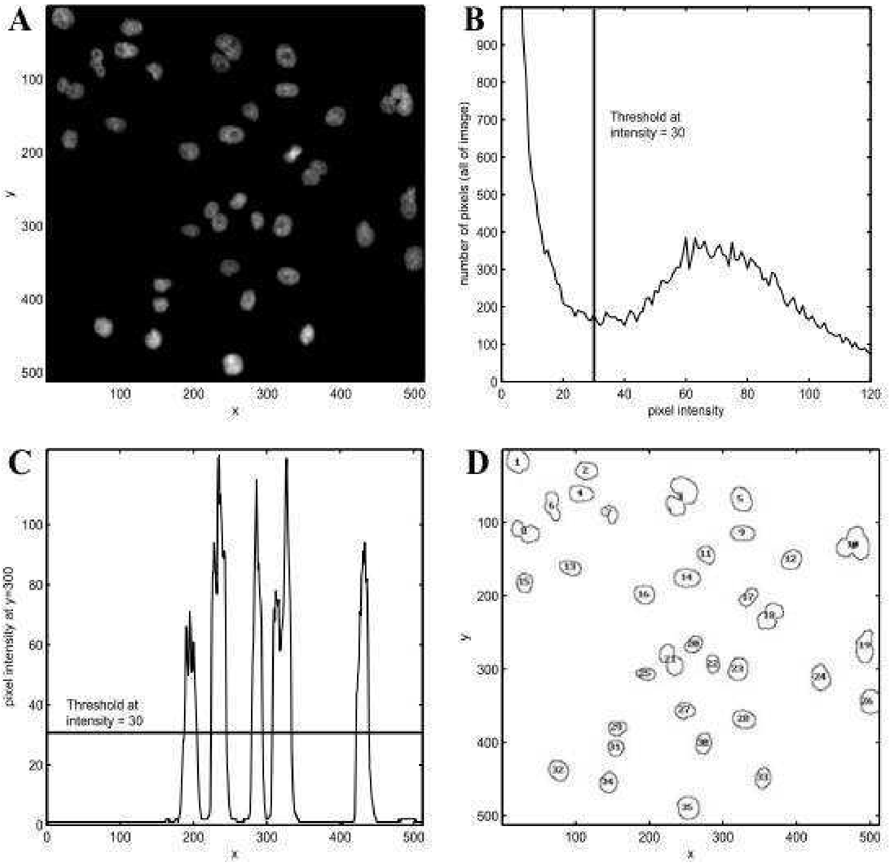


**Figure 1.8:** *From [218]: A: Fluorescence stained nuclei of cultured cells. B: Image histogram of A. A threshold is placed where the histogram shows a local minimum. The vertical line corresponds to a threshold at intensity* 30*. C: An intensity profile along the row* $y = 300$ *of A, with the intensity threshold represented by a horizontal line. D: The result after thresholding and labeling of connected components. All nuclei are not separated by thresholding.*

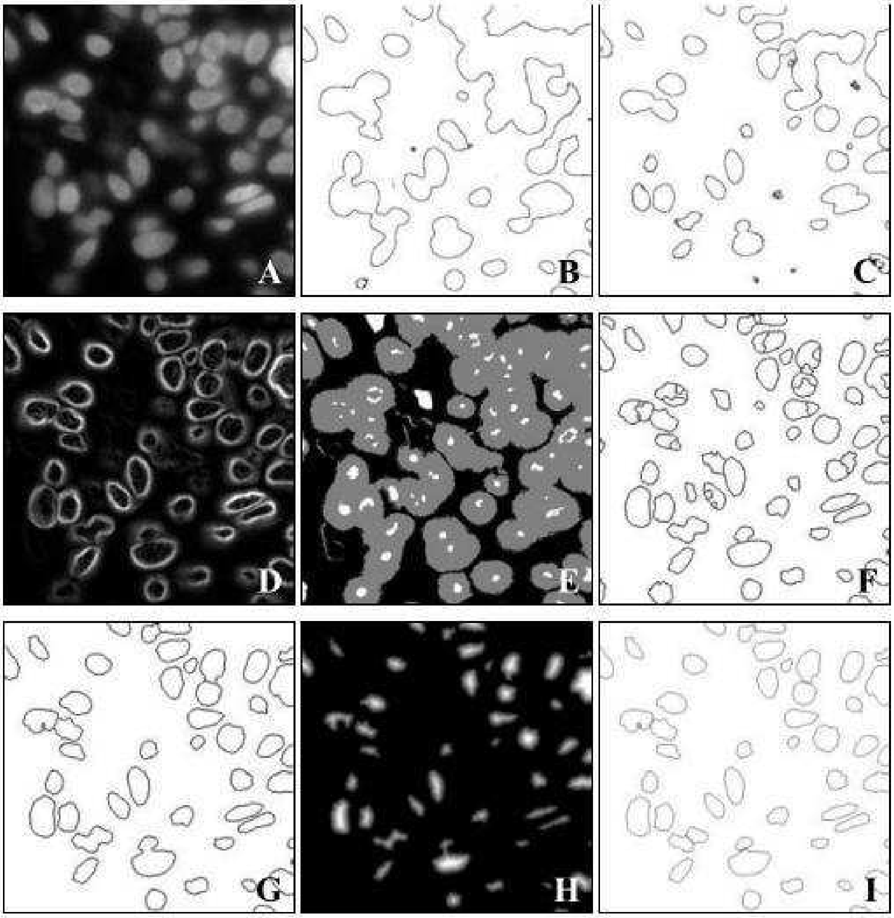


**Figure 1.9:** *From [218]: A: Part of an original 2D fluorescence microscopy image of a section of a tumor. B: Result after thresholding at intensity 60. Most objects are detected, but a lot of background is also above the threshold. C: Result after thresholding at intensity* 100*. Only a little background is above the threshold, and some nuclei are nicely delineated, but many are not detected at all. D: The gradient magnitude of A. E: Object (white) and background (black) seeds found by the extended h-transformation of the original image and the gradient magnitude image, respectively. Small components were removed from the background seed. F: Result of seeded watershed segmentation; some objects are over-segmented. G: Result after merging seeded objects based on edge-strength. Poorly focused objects are removed in this step. H: The distance transform of the objects in the segmented image provide information on object shape. The brighter the intensity the further away from the background, or a neighboring object, the pixel is. Watershed segmentation of this image separated clustered objects. I: Final segmentation result based on intensity, edge, and shape information.*

processing, feature extraction is a special form of dimensionality reduction. When the input data to an algorithm is too large to be processed and it is suspected to be notoriously redundant (much data, but not much information) then the input data will be transformed into a reduced representation set of features (also named features vector). Transforming the input data into the set of features is called feature extraction. If the features extracted are carefully chosen it is expected that the features set will extract the relevant information from the input data in order to perform the desired task using this reduced representation instead of the image-sized input. In the context of migrating cancerous cells, features could pertain to the whole population of cells or to an individual cell, and could measure aspects of different nature in the image; as shall be the order of presentation of some of such features in this section.

### 1.4.1 Object counting

Cell recognition and counting in microscopic systems is an attractive and challenging task due to the presence of debris, high noise, and the difficulties of adapting available image segmentation approaches. To evaluate cell distribution, types and the migration mode it is required to count either the total number of cells or that of the cells which have migrated during end-point analysis. A correct identification of these objects usually requires a combination of the segmentation and post-processing stages described above, but sometimes a bespoke object recognition and counting procedure e.g. when a count is needed before the application of those stages or is necessitated by the quality of the image or of the cells.

Several authors discuss the counting of cells in microscopy images. A cell count as a byproduct of segmentation, as mentioned before, is usually the obvious method of choice. [232] detail a process of binary-thresholding the image and counting the number of objects thus obtained, given an image of non-adhesive cells. The indirect approach to the problem of counting cells uses globally estimated features as intensity of density or density of color to approximate the cell quantity. In [164], a system to measure the relative cell quantity in culture plates makes use of total fluorescence after background fluorescence reduction as a measure of a number of cells per plate. In [130], cell quantity is estimated by dividing a cell cluster area by means of a cell area. Selinummi et al. [177] exploit organelle-selective marking to achieve automated segmentation to identify image regions composed of cells stained by a given biomarker. Korzynska [103] works on phase-contrastmicroscopy with unstained cells, manually classifying stem cells into three morphologies and combining different segmentaion approaches for each, in order to ascertain the number of cells. This type of method is evidently only applicable if the number of cells is small and manual labeling is possible.

A template matching approach, also exploiting selective-marking, is used by Kachouie et al [96] to separate live and dead cells, stained in different colours, and then correlation maps are developed for each by template matching with a cell-sized disc, the maxima indicating cell centres (Fig. 1.10 ).

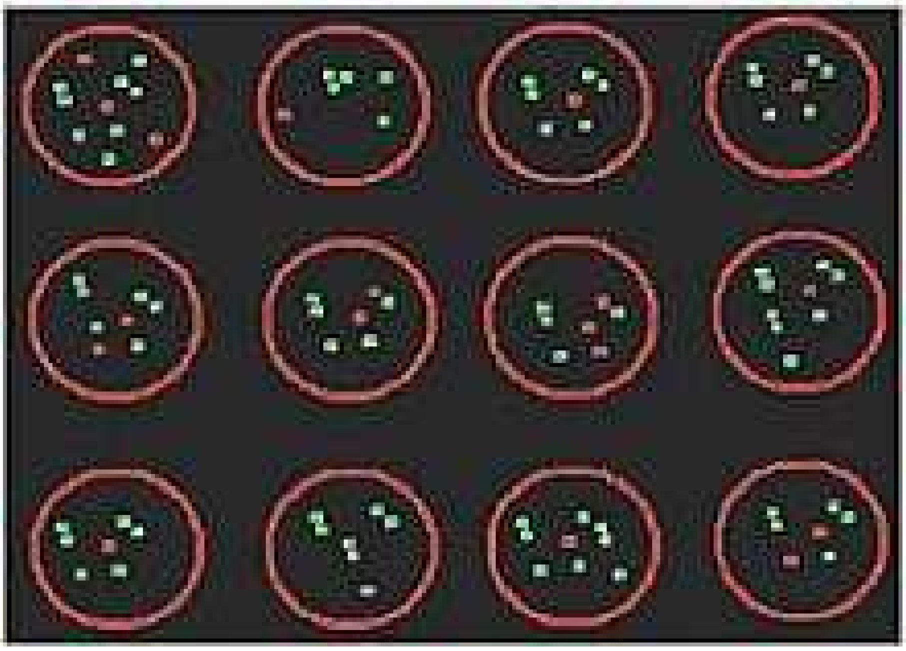


**Figure 1.10:** *From [96]: Separating and counting stained, i.e. living (green) and dead (red), cells in microwell arrays.*

Faustino et al [60] also use histogram-thresholding on fluorescent cellular blobs with the calculation of the resulting centroids, and present an interesting comparison with manual counting performed by a panel of experts, showing the significant diversity in manual counts and the difficulty of validating automatic counts. Fig. 1.11 shows their method of segmenting out individual cells, and Fig. 1.12 the result of counting these cells.

A different method for object-discovery in greyscale images is proposed by [74], based on $a - contrario$ [51] methods. They build a statistical model to predict the detectability of a spot on a textured (grainy) background and use binary hypothesis testing to decide whether a spot, possessing a noticeable contrast, is present or not in a given realization. This approach is yet to be explored by the wider community.

Recently a *pixel patch* strategy has been formulated as described in [146]. The process involves two stages: preprocessing and classification. The major task of pre-

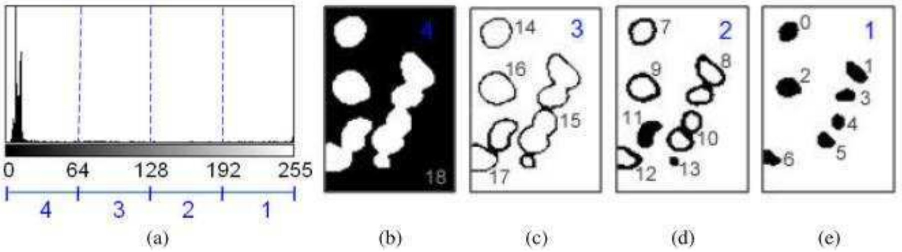


**Figure 1.11:** *Histogram partition and connected components detection: a) histogram, b), c) d) and e) bitmaps representing the intervals 4) [0, 63], 3) [64, 127], 2) [128, 191], and 1)[192, 255], respectively. The numbers besides the cells are the label of the connected component. Note that the higher is the label value smaller is the luminance of the component.*

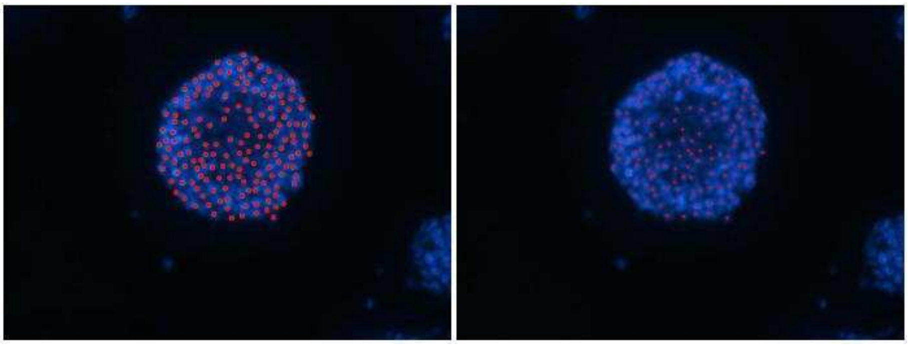


**Figure 1.12:** *From [60]. Results of the manual (*163 *cells counted) and automatic (*143 *cells detected) counting, images on the left and right respectively.*

processing is to derive a representation of cells which makes subsequent classification computationally effective and insensitive to environmental changes by providing the classifier only with information essential for recognition. In the classification stage, a neural network is trained to determine if a pixel patch contains a centered cell body. This is done with pixel patches represented by feature vectors derived in preprocessing. Although Long et al [123] simplify it by not learning the not-cell patch, and following the calculation of the confidence map by a local maxima detection to designate cell centres. Higher accuracy rotation-invariance are obtained by Theis et al [200] [201] using unsupervised independent component analysis with correlation comparison. In order to account for a larger variety of cell shapes, they also propose a directional normalization.

### 1.4.2 Population movement measurements

As mentioned before, the global migration property of a cell population usually refers to its ability to colonize its neighborhood or the entire culture medium. This ability can be easily monitored by analyzing phase-contrast time-lapse images or by end-point analyses of fixed and stained materials. The net increase in the total area covered by the cells is evaluated by segmenting the surface occupied by cells from the background. In addition, this segmentation process enables the migration front to be identified, enabling the measurement of the rate of advance and/or the distance covered by cells in the migration front [29]. Fig. 1.13 shows their experiment with cell colonization of a wound created in a confluent monolayer of cells at time 0 ($t0$). This colonization ability is then evaluated at $t1$ either by determining the net increase in the total area covered by the cells or the rate of advance of the edge of the wound. These measurements are evaluated by segmenting the surface occupied by the cells from the background (cf. hatched areas).

### 1.4.3 Cell trajectory movement measurements

A particularly useful feature in the study of cell locomotion is the reconstruction of the trajectory covered by each cell from a frame to the following one in a sequence of images. The high processing demand for extended-time studies of large cell populations rules out the use of manual or computer-aided interactive tracking. Fully-automated techniques are required. Methods for automated object tracking mainly involve two different approaches: tracking by detection and tracking by model-evolution.

The first approach performs object detection and inter-frame data association

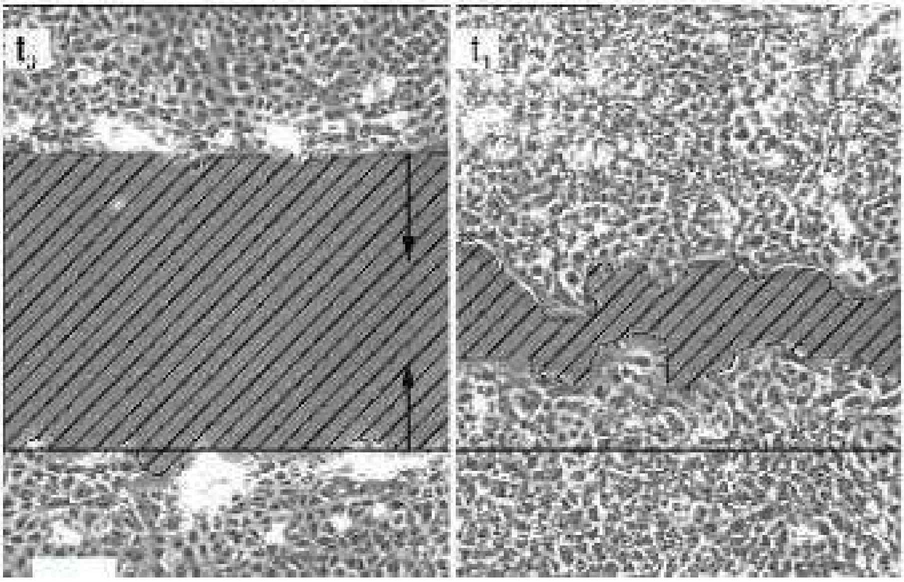


**Figure 1.13:** *From [29]. Front migration example: cell colonization of a wound.*

in two independent stages. This approach is effective when the objects are well-separated, but faces ambiguousness if the objects undergo close contact, and when the detector produces split/merged measurements. De Hauwe et al. [86] provide an example of frame-by-frame segmented object tracking by segmentation followed by inter-frame object-pairing. Khan et al [99] addressed the issue of split/merged measurements, but under the assumption that the number of objects does not change.

The second approach involves the creation of mathematical models, either appearance or shape models, which are fitted to the objects and are evolved over time to follow the object movements. This category encompasses a large spectrum of techniques with varied capabilities. The parametric active contours (e.g, snakes [97] [228]) and mean-shift [47] models have been explored in the past for tracking multiple migrating cells under phase-contrast microscopy. Debier et al [47] considered a somewhat simplified problem of tracking only the centroid positions, but not the boundaries, of the cells, which permits a mean-shift-based model to be used to establish migrating cell trajectories through *in vitro* phase-contrast video microscopy. Fig. 1.14 shows tracking through frame-by-frame stochastic model filtering. The numbers at the top-left corner are the frame indices. Cells were manually-labeled. Those bearing labels 2 and 10 are partially overlapping in frames 65 – 67. Cells 6 and 12 are closely passing each other.

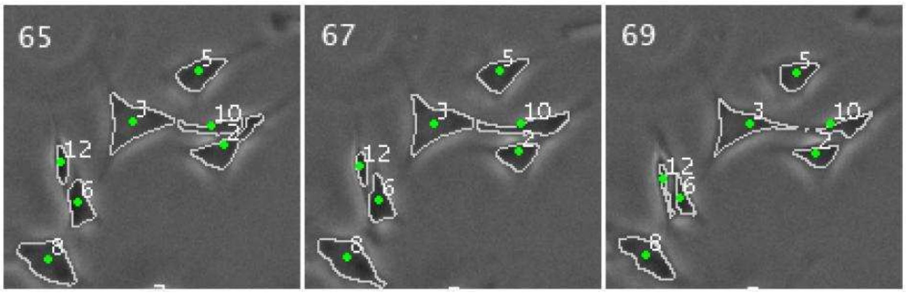


**Figure 1.14:** *From [116]: Tracking closely contacting and partially overlapping cells.*

### 1.4.4 Measurements related to rare cellular events

When cell cultures are observed for longer periods of time (a day or more), it thus becomes possible to detect less frequent cell events, such as cell division or death or migration-mode-transitions. A very interesting discussion about a rare metastasic event triggered by the collective behaviour of the cell population and of the environment is found in [128]. The literature also reports related studies in the fields of neural and clonal development [4].

### 1.4.5 Shape and cell morphology

Historically, visual inspection was the only way to distinguish different cellular patterns in morphologies of objects in microscope images. However, visual classification is always time-consuming, subjective, and inconsistent between experts. A major goal in microscopic image processing, one that ties in with pattern recognition, is to develop systematic approaches to describe cellular shape, including building classifiers that can recognize them. Once the objects of interest are segmented, a set of shape features can be extracted [45]. Cell shape descriptors in 2D environments include area and perimeter [185]. The complexity of the cell shape can be expressed by means of a circularity index (equal to $4\pi Area/Perimeter^2$). Zaman et al. [234] used this “cell aspect ratio”, i.e. an index of shape elongation defined by the ratio of the length of the major axis by the minor axis of the ellipse that best fits the object, to characterize the 2D projections of the shapes of cells cultivated in a 3D gel. This index enabled amoeboid cells, which presented a spherical morphology and an aspect ratio of 1, to be distinguished from mesenchymatic cells, which showed an elongated morphology and a greater ratio.

Shape information from segmented cell contours and various ratios of cell cytoplasm and nuclei in Kim et al [100] are transformed into a polar representation and reduced through $PCA$ and classified through a neural network.

In their work on grouping natural images Ren and Malik [169] introduce some features for comparing two regions, based on classical Gestalt principles of grouping including proximity, similarity, good continuation ( curvilinear continuity ), closure as well as symmetry and parallelism, which since they are *ceteris paribus rules*, i.e. that they distinguish competing segmentations only when everything else is equal, form grounds of comparison with ideal class examples.

Lai et al [107] define a set of morphological features and a expert-learned set of their standards for each type of cell that could be encountered in hepatocellular carcinoma images. These features include the *nucleocytoplasmic ratio*: nuclear density, nucleus-to-cytoplasm ratio, and cell-size; *nuclear irregularity*: circularity as defined above, area irregularity calculared using four intersecting points between a nucleus and its bounding rectangle and the contour irregularity of the nucleus defined by curvature at sample points on the contour; *hyperchromatism* i.e. excessive pigmentation in hemoglobin content of erythrocytes by the average intensity of nuclei and the ratio of the number of bright and dark spots found through morphological top- and bottom-hat operators [214]; *nuclear size*: the number of pixels covered; *anisonucleosis* i.e. difference among nuclei, described by standard deviation of nuclear size and the difference of extreme nuclear sizes; and finally *nuclear texture* using three features of the Grey Level Co-occurrence Matrices (GLCM) [81] [84].

Shape representation and description generally looks for effective ways to capture the essence of the shape features that make it easier for a shape to be stored, transmitted, compared against and recognized. However, shape representation and description also considered a difficult aspect since shape is often corrupted with noise. Several attempts have been made in order to find more discriminatory and more efficient shape representations, such as chain coding [194], Fourier signature [98] etc. Salih et al [175] discuss one such approach. The crux of their method is to neglect noise distortions by decomposing the $2D$ object boundary into sequence of straight-line segments (lengths and directions), which lead to generate an approximate representation of the original boundary.

There are two bases that can be exploited to formulate a function of contour [208], the *symmetry* and the *periodicity*. According to Djemal et al [38], if the contour of an object is symmetrical, the orthogonal distance from a point on the contour to the axis of symmetry is an example of the first contour function. The considered example of the second function uses representation of the contour in polar coordinates and is the description of the contour by its curvilinear abscissa and the tangent to the contour at any given point. Using these functions, they derive several descriptors such as chord, extremities and inscribed and circumscribing circles, which are then classified using a radial basis functions neural network [15].

The final goal in automatic image analysis is to make a decision with respect to image contents, for instance, to grade the severity of a disease from a cell culture image, to assign semantic labels etc. To make the final decision, by appliying decision rules, learning algorithms or similarity measures, an image representation is needed. Clearly the set of all pixels in the image is an amount of data that cannot be efficiently included in an “on-line” decision process, and pixel data is massively redundant and complex in its internal correlation. The manner to use the data is therefore to measure pertinent quantities, such as the amount of front propagation, and to summarize it in quantifiable descriptors or features.

## 1.5 The problem at hand

Due to lethal consequences of metastatic spreading of cancer, understanding and controlling the processes underlying the formation of metastases is a major challenge, remaining largely open. Several modes of metastatic spreading(letting aside surgical dissemination) were identified: (1) transport in lymphatic circulation, (2) transport in blood circulation, and (3) a mode involving a specific migration mechanism, the *amoeboid* migration [63].

Taking place at the cell scale, it appears less pervasive than the first two ones, where circulation-facilitated transport spans the whole organism. However it is less dependent on the anatomical features of the location of the tumor and is a candidate for the early events of the metastatic spreading,before metastatic cells reach the lymphatic or the blood circulation. It might well be an essential preliminary step common to all metastatic processes [31].

### 1.5.1 Biological background

Early events involved in the escape of a cancer cell from the primary tumor pose a difficulty because they are rare events, too rare in fact to be easily observed or experimented in varying conditions. Only a small fraction of tumor cells provides the seeds for secondary tumors. Accordingly, experimental protocols are restricted to indirect investigations, mainly genetic and biochemical analyses of metastatic cells compared to those of the primary tumor [226] or statistical tracking of the number, location, and genetic lineage of secondary tumors [1]. An increasing number of experiments focus on the biochemical analysis of the surrounding microenvironment [197], the morphological signature of potentially metastatic cells [212] and the reproduction *in vitro* of the *epithelial-mesenchymatous transition* ($EMT$) and the *mesenchymatous-amoeboid transition* ($MAT$) (Fig. 1.15) which affect the morphology and the proliferative and migratory capacities(amoeboid migration) of cells of epithelial origin [129]. These complementary experiments have shown that the metastatic process involves jointly genetic determinants (accumulation of specific mutations, biochemical factors) triggering new pathways or switching existing ones, leading to modifications in the cell state and metabolism), and requirements about the state and geometry of the microenvironment, that is, the extracellular space and matrix of the tumor cells.

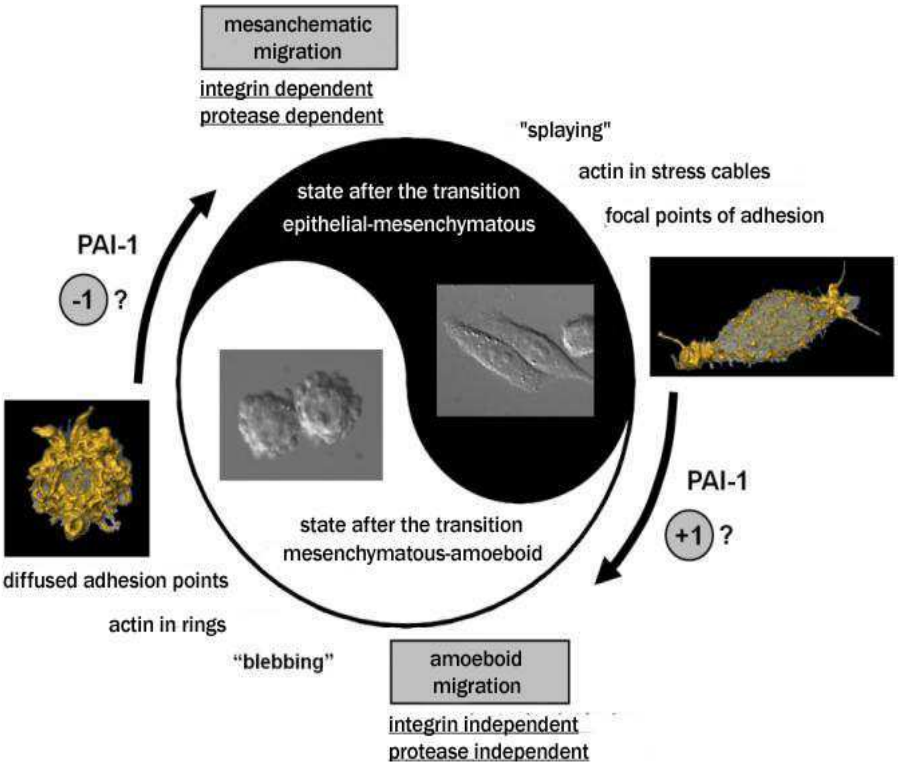
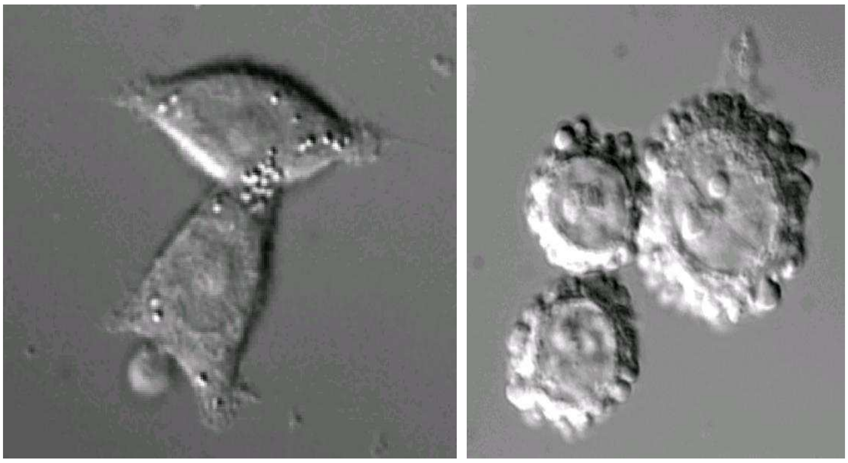


**Figure 1.15:** *The Yin and the Yang of migration of a cancerous cell [31].*

Considering epithelial cells in Fig. 1.16, a first transition toward a cancerous state is observed, originating in accumulating mutations and leading to the so-called mesenchymatous state (Fig. 1.16 a)). In this state, cell-cell junctions are no longer established and the epithelium is destabilized. This state has moreover a strong proliferative capacity, hence the transition to this mesenchymatous state is generally associated with the appearance of a well-defined tumour [203]. In invasive epithelial tumors, it is the default state of the cells [68]. The mesenchymatous-amoeboid transition (as experienced by the right most cell in 1.16 a) is likely to play a key role in early metastatic escape. A second transition may occur towards the so-called amoeboid state (Fig. 1.16 b)) identified by a specific and persistent "blebbing" morphology (round shape with dynamic actin rings visible at the cell periphery). This mesenchymatous-amoeboid transition is associated with a change in adhesion properties. Due to its peculiar features, a blebbing cell can move fast and progress by exploiting interstices of the substrate with no need of matrix proteolysis. Accordingly, amoeboid migration is a very efficient mode of migration in a tissue, encountered in normal conditions during some developmental stages [203]; in a pathological context, it was suggested as a privileged mode of metastatic migration [63, 16].



(a) (b)

**Figure 1.16:** *Epithelial cancer cells (colon cancer). (a) Mesenchymatous state, responsible for the destabilization of epithelium and prone to proliferation. (b) Amoeboid state characterized by a blebbing morphology and by modifed adhesion leading to a special migratory ability.*

Recent observations *in vivo* hint at a key player in amoeboid migration, metastases, and more generally cancer progression: the **Plasminogen-Activator-Inhibitor protein of type 1**, hence forth termed by its acronym $PAI-1$. It is an ubiquitous species involved in several pathways and functions, among which some aspects are relevant for metastatic process. It is found in the surroundings of the most invasive tumors [158] [224] and considered as a marker of bad prognosis [94, 58, 32]. Strikingly, when cancer cells are placed on artficial substrates with high concentration of matrix-bound PAI-1, they experience the above-mentioned mesenchymatous-amoeboid transition [129].In this respect, matrix-bound PAI-1 can promote cancer cell migration, at least *in vitro* [63]. Moreover, these experimental results, presented as the horizonatal bar graph in Fig. 1.18, indicate that the mesenchymatous-amoeboid transition is not due to some mutations but is rather a dynamic transition between two different states of the cell, controlled by its environment.

### 1.5.2 Experimental objectives

The biologists we are partnered with investigate the effects of $PAI - 1$, on the behaviour of migrating cancerous cells [210]. These cells are clones of the cell that initiated the cancer, having acquired certain characteristics allowing it to divide indefinitely and be able to metastase i.e. to proliferate and migrate. These cells adopt one or both of two types of migration: *mesenchymatic* and *amoeboid* [63]. Amoeboid migration is fast and is usually responsible for metastasis, while mesanchematic migration results in proliferation within the same tumour. During their migration process, amoeboid cells first adopt an amoeboid form referred to as *blebbing*. This allows them fast movement governed by weak and short-term interactions with the extracellular matrix. Due to their high deformability, they can slip through the extracellular matrix without deteriorating it. On the contrary, mesenchymatic cells become polarized and choose a migration front along which form new points of adhesion to the extracellular matrix, while those on the posterior part of the cell detach themselves from it, propelling the cell along.

Changes in migratory behaviour have been observed experimentally and associated with phenotypic or morphological switching in various situations, such as the migration-proliferation dichotomy of the epithelial-mesenchymal transition or the mesenchymal-amoeboid transition of cancer cells in the extracellular matrix [70] [35]. Fig. 1.17 shows the three morphologies: blebbing [59] and spread, pertaining to amoeboid and mesenchymatic migrations respectively as well as smooth round pertaining to the intermediate stage. As shown in Fig. 1.18, cell morphology changes greatly according to the microenvironment. In particular, the presence of $PAI - 1$ discourages cells to change into the spread morphology, this effect eroding with time as the chemical is gradually consumed. The difference in morphology as function of different substrates is the greatest at 6h and diminishes later. The understanding of this migration process could help to arrest the development of cancer and increase the chances of a cure.

In an experiment, the goal is to determine what cells on any sample image are in which stage of the metastatic process. This general objective spurs us towards more concrete sub-objectives of being able to recognize parts of the image as cells (the object of literature in Sections 1.4.2 to 1.5.1), then recognize the cells into differentiable categories according to their metastastic stage (requiring information such as that mentioned in Sections 1.5.4 and 1.5.5), and through the nature of the image data and the image processing operations implicated necessitates the use of techniques briefly mentioned in Section 1.4.1, thus employing practically the whole

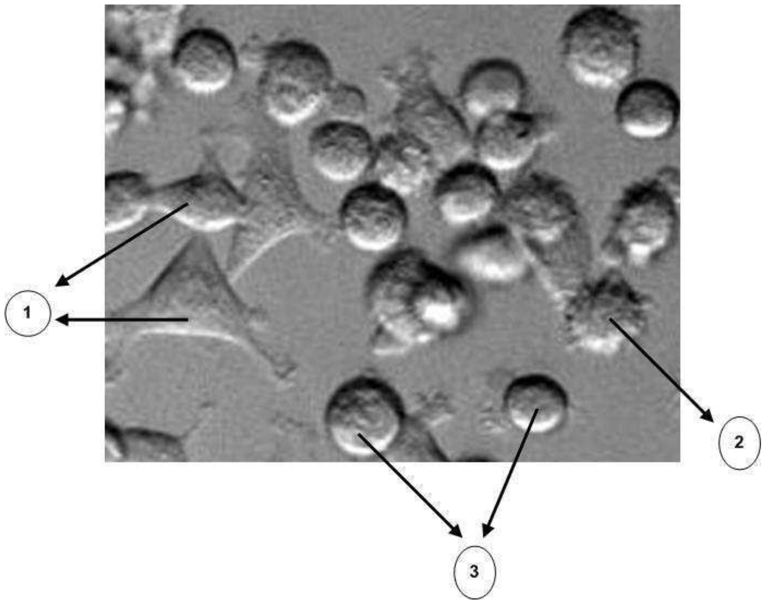


**Figure 1.17:** *The three types of cells: (1) spread or mesanchematic, (2) smooth round or transitory, and (3) blebbing round or amoeboid cells.*

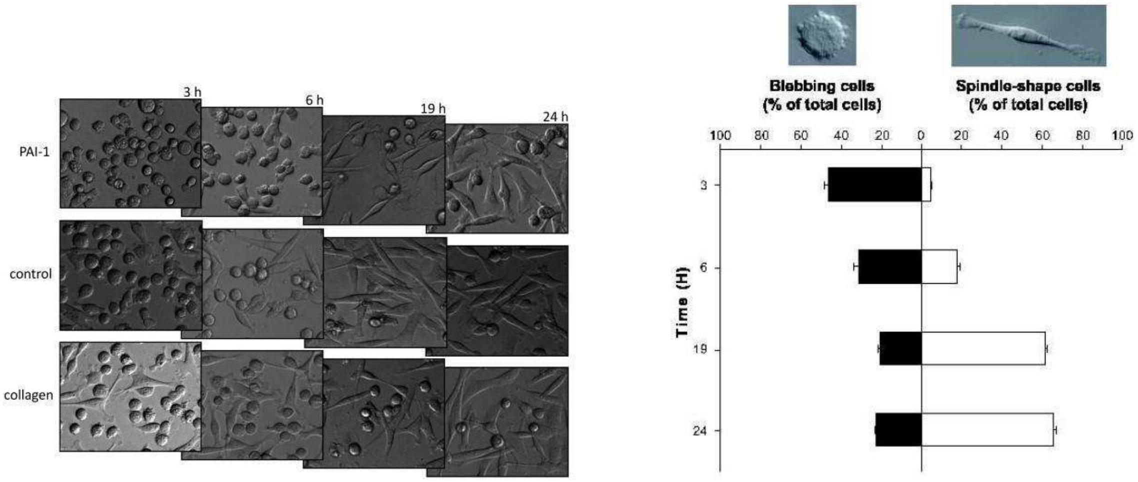


**Figure 1.18:** *Qualitative and quantitative study of general cellular morphology (round vs spread) as a function of the microenvironment: (left) time-lapse photography of* $MDA-MB-231$ *cells seeded on* $PAI-1$ *or collagen substrate; blebbing cells becoming spread show the reversibility of the mesenchymatous-amoeboid transition (*$MAT$*). (right) the proportion of blebbing and spindle-shape (spread) cells seeded on a* $PAI-1$*-enriched microenvironment is shown at successive time points (*$3$, $6$, $19$, $24$ *hours). The proportion of blebbing cells (horizontal axis) decreases in favour of spread morphology indicating the reverse* $MAT$ *[31].*

slew of the image processing tools that have been mentioned. The study *sic* of individual cells synthesize into that of an entire image in that the numbers and thus proportions of each of the 3 types of cell could be obtained, and the process of metastasis (e.g. the reverse-$MAT$ in Fig. 1.15) could be analysed.

This can be done manually by an expert, with the advantage of the confluence of the cell-recognizing/counting and the category-attribution processes into one decision-making process of the kind “Is this a cell, and if it is, then what sort?”. But is a tedious job, and one that requires constant attention and mental focus. $DYNAMIC$ determined that cell labeling accuracy declines significantly over extended spans of time, due to the repetitive and monotonous nature of the work. In addition, a human expert has to be engaged; and in the case of the data we shortly present the experts took 3 weeks to accomplish it.In the context of this endeavour, our goal is to develop automatic image processing techniques that could optimistically replace the human intervention, while attaining comparable accuracy. This would provide a particularly useful tool for biologists studying the evolution of cancer under varying environmental conditions.

Before we proceed to the actual processing ensemble we will briefly present the data and its acquisition in the following, since the following chapter continues with a more detailed insight into the data and its characteristics.

## 1.5.3 Materials

### 1.5.3.A Cells and cell culture

To study the behavior of cancer cells as a function of the microenvironment, two cell lines, each very invasive, of epithelial morphology and different origins are used: line $SW620$ ([Ref. CDC-227], ATCC, Rockville, MD, USA) from a metastasis of a human colorectal adenocarcinoma located at the lymph node, and the $MDA-MB-231$ ([Ref. HTB-26], ATCC) derived from a pleural metastasis of human breast adenocarcinoma.

Cells were grown in Leibovitz’s $L-15$ Medium (Invitrogen, Carlsbad, CA, USA) with GlutaMAX supplemented with 10% fetal calf serum ($FCS$) (Sigma, St. Louis, MO USA), $100U/mL$ penicillin and $100U/mL$ streptomycin (Invitrogen). Cultures were selected with $400\mu g/mL$ $G418$ to generate stable cell lines and maintained in exponential growth in a humidified atmosphere at 37ř$C$ without carbon dioxide. The passages are performed twice a week at a seeding density of $30000 cells/cm^2$: one at day 3 and day 4 to the other. The medium was renewed at day 2 when the transition is expected to day 4 and all experiments are performed at day 3. Cells were fed every third day and only used at passage four to seven for the experiments. All experiments were performed three times on each of the three cell lines cultivated in usual conditions (three days in Leibovitz's $L-15$ Medium completed with $FCS$).

### 1.5.3.B Data and its acquisition

In the course of this work, solely cell line $MDA-MB-231$ has been studied in two situations: a pro-migratory $PAI-1$ environment ($20mg/cm^2$) and a non-permissive collagen control ($20mg/cm^2$) environment. The material includes two series of greyscale images of 1388 × 1040 pixels in TIFF format, the first - *time effect* - studying the effect of time on cancer development under the two substrates, comprising 79 and the second - *dosage effect* - studying the effect of $PAI-1$ dosage, comprising 63 images. To give the readers an idea of cellular density, as we shall see in Chapter 2, it is about $460-470$ cells/image, depending on the expert.

**Figure 1.19:** *Zeiss Axiovert200 microscope with incubation chamber for phase-contrast microscopy, capable of Z-plane photography through the APOTOME system.*

The cells were studied with a $Zeiss\ AXIOVERT$ 200 inverted microscope, which is built for examination of tissue culture flasks, Petri dishes, microtiter plates, etc., in transmitted and reflected light. It can be used for bright field, phase contrast, differential interference contrast, and epi-fluorescence techniques. The apparatus was coupled with a Siemens $CCD$ camera and digitized on an image processing unit at a final magnification of 5000× and $0.05mm^2$ per pixel.

#### 1.5.3.C Computational resources

The experiments were set up on a *Pentium D* machine, with a $3.2GHz$ *CoreDuo* processor and $1GB$ memory. The bulk of the work was performed on MATLAB 7.0 ($http://www.mathworks.com$) doted with the Image Processing Toolbox, and for the purpose of computational speedup some programs were developed in C and interfaced with Matlab. Classification of cells also uses the statistical software R ($http://www.r-project.org/$).

## 1.6 Summary and conclusion

Biological discovery is advancing toward the use of high-throughput experimental approaches for applications in genomics, proteomics, drug development, tissue engineering and stem cell research. A recent focus area is chemical-induced changes in the cellular microenvironment that play a role in directing cell fates. The study of how microenvironment patterns regulate migration, proliferation, and apoptosis requires the use of non-fluorescent phase-contrast microscopy to record the cellular responses over an extended period of time, which routinely produces large datasets with low signal-to-noise ratios.

This chapter serves as an appraisal of the kind of problems today's biologists, cytologists and histologists contend with, and what kind of aid and in which areas of the different stages of their works do imaging technologies and the technologies of processing those images bring to the table. Many of these will be revisited in the course of the thesis, but here we presented an overview of image techniques such as acquiring the image, correcting the acquisition biases through low (pixel)- level methods, zeroing-in on the salient parts of the image, namely the cells, separating them from the rest of the image as identifiable individuals - image segmentation, obtaining various measurements on those individual objects - feature extraction, and also on the entire population of those individuals as an interactive microcosm of the latter - cell counts and population growth and movement measurements. The pertinence of these techniques is embodied in the problem at hand; it is a celebration of cellular image processing methods, from denoising and illumination correction to individual cell segmentation to an extraction of their phenotypic features to studying the overall populational transition of cancer stages brought about by changes in their microenvireonment.

This chapter therefore lays the foundation on which the work we present in the following chapters could be established. Once we have shown *why*, and equally importantly, *how* these images have been obtained, we are prepared to broach the objectives laid out in Section 1.5.2. Hence, Chapter 2 will be dedicated to the first objective, i.e. gathering the number of all cells over an image, but also to the various issues with our phase-contrast microscopy and their mitigation in what could be called pre-processing to the actual processing-for-our-objectives. Chapter 3 will consecrate itself to the crucial problem of separating cells from the image background and among themselves. Finally Chapter 4 brings the path full circle by synthesizes the results issued from Chapter 2 - the total number of cells, and chapter 3 and 4 - the silhouettes of individual cells, through the study of the shape and texture of these individual cells to assign them to the three classes in Chapter 5 , finally arriving at a prognosis of the cell sample represented by the image as a function of the number of cells obtained in Chapter 2 in each class.

# 2

# Pre-processing and Cell Detection

As mentioned before, the objective of an individual experiment designed to offer prognostic of a cell sample is to determine which and how many cells in which stage of metastasis. However, under the experimental setting that generates data for this work, where all cultured cells are in various stages of metastasis, the goal is to determine the proportion of each migratory mode represented on an image, which will enable biologists to determine the preponderant migration route the cells in the image would take. Either of the objectives mandates determining the proportion of each type of cells, and therefore determining a global cell count as well as counts of each type of cell. Cell counting is usually done manually by an expert, but it is a hard and tedious task. Our goal is to develop automatic image processing techniques that could replace the human intervention, while attaining comparable accuracy. This is a particularly useful tool for biologists studying the evolution of cancer under varying environmental conditions.

In this chapter we will focus on *cell detection*. By cell detection we refer to a process that provides us the number and the locations of cells. The main focus of our work, directed toward achieving the best possible cell detection performance, is to develop a filtering step allowing optimal adaptive estimation from the local clutter. This process of cell detection and the validation of that detection has consequences on the statistics obtained on the three classes of cells, since a preferred detection algorithm would produce equal amount of error across the cell classes. Cell detection is the *first step* of a larger chain of processes, including cell segmentation and cell classification, the latter all depending on the accuracy of this primordial step.

However, before embarking on cell detection, the various intrinsic difficulties in the data must be overcome in order to allow for a set of processing steps that would be less fault-prone and in fact even possible only as a result of this pre-processing. As we shall shortly see, a first step of image segmentation has also been incorporated into the set of pre-processing steps for reasons of data dependency. Fig. 2.1 gives an overview of the experimental aspects of the chapter, from pre-processing through to cell detection.

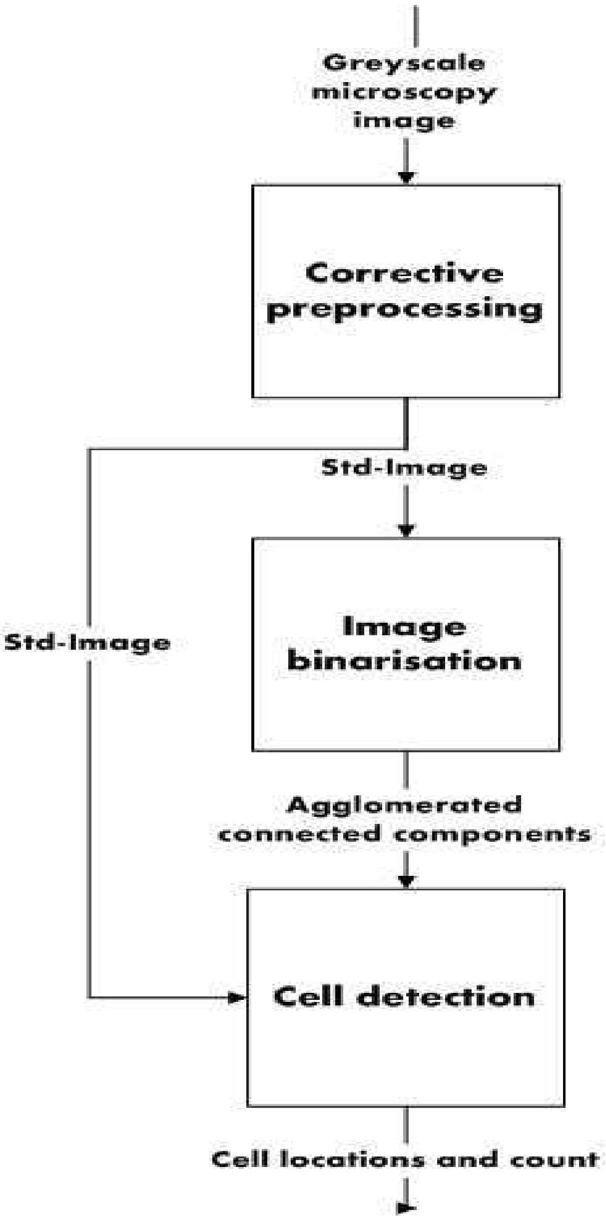


**Figure 2.1:** *Schematic of the experimental processes involved.*

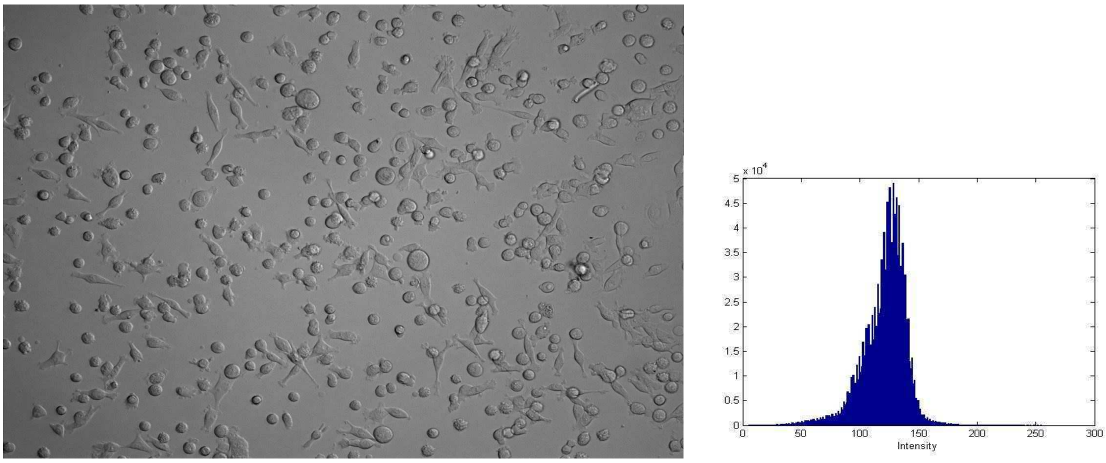
## 2.1 Corrective pre-processing

Let us recall some of the ideas presented in Chapter 1. We saw how the conditions of image acquisition could be removed from the ideal through culture visualisation constraints and the limitations of phase-contrast microscopy e.g. the amount of exposure, the agglomeration of cells, occlusion and lighting conditions at the moment of acquisition. Then there was the mention of several low-level image processing methods that could help alleviate some of these issues post-acquisition and pre-main-processing. Similar conditions are encountered as this section describes, and methods of similar classes are employed to mitigate them. Since this is a pre-requisite to cell detection and segmentation, we call this step pre-processing.

Pre-processing, as the name suggests, alters the content of the image and makes it more suitable for the following image analysis steps. The aim of this step should however not be to make the images “look nice” since by doing so, one might risk throwing away useful information, but rather to make them more suitable for the next steps to be applied. Let us look at the issues and their pre-processing solutions.

### 2.1.1 Data-induced challenges

There are several constraints that the data impose on processing, which make our fundamental task more complex. Part of the difficulty comes from the fact that the image acquisition process is not standardized, meaning that focal length and lighting conditions are not uniform for the entire data. Let us take a look at each of these challenges one by one, followed by the solutions used to palliate them:



(a) (b)

**Figure 2.2:** *a) A sample image in* $PAI - 1$ *environment, b) its intensity histogram.*

1. **Low contrast** The contrast of greylevel intensities between the cell and the background is very low (Fig. 2.2 *a.*). This is evident from the intensity histogram (Fig 2.2 *b.*). The histogram of the cell images are usually unimodal and very narrow, reflecting the negligible separation in average intensity of the cell and background, and the feeble dynamic of the image.

2. **Inhomogeneous illumination** The images present a global illumination gradient, which means that the greylevel of the background, i.e. the non-cell component of the image, varies across the image (Fig. 2.3). This is due to the

fact that cells are suspended in liquid with light sources above and below them that microscope operators adjust to their convenience. This effect is rendered even more complex by the fact that the direction and intensity of the light varies from one image to another. In many images this global luminosity is sourced from a point outside the image borders, while in others it is centered on the image itself.

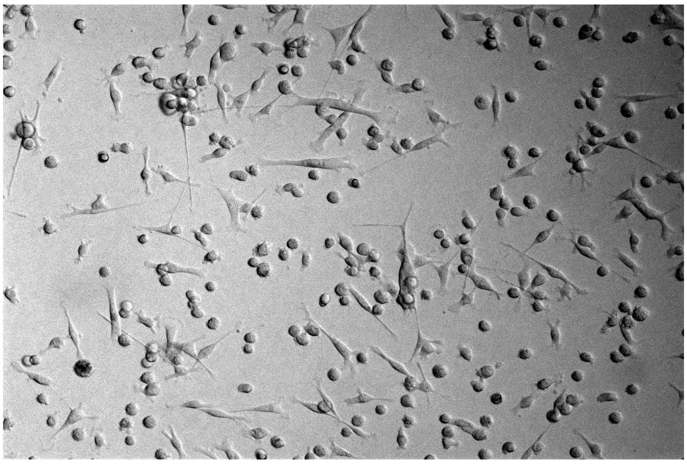


**Figure 2.3:** *The illumination gradient visible from bottom left toward top right.*

3. **Isotropic exposure to lighting** In the absence of pigmenting, light from sources of the microscope offers the discriminating element between the cells and the background by illuminating or not, parts of cells, and causing shadows along others. This introduces bias in image processing, particularly for segmentation algorithms, since edges that represent cell features are offset from them.

4. **Varying cell visibility** Round cells are more prominent than elongated ones (Fig. 2.4) because their cellular matter is contained in a smaller area, giving them more height than their spread counterparts. This creates problems in the cell detection, where the presence of round cells could overshadow that of spread neighbours, leading to fewer cell being identified than really exist.

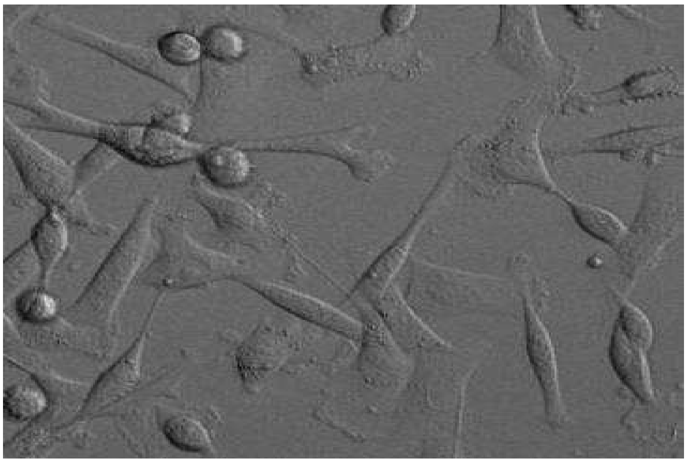


**Figure 2.4:** *Rounds cells are usually more prominent, making spread cells more difficult to detect.*

5. **Cell agglomeration** These living cells undergo the processes of anchorage, adhesion and de-adhesion as they do in their natural milieu. This causes agglomerates of cells to form. The top-view microscopy therefore presents these cells as overlapping each other (see Fig. 2.5), which renders the problem of discerning individual cells difficult even for the human eye due to partial occlusion.

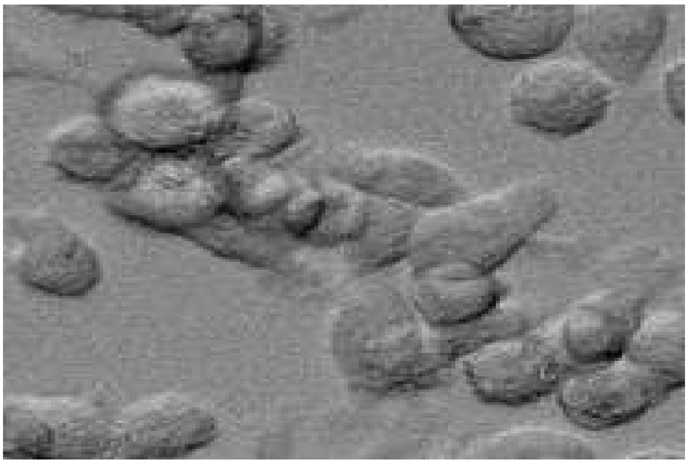


**Figure 2.5:** *An agglomerate of overlapping cells.*

The following two subsections discuss the two preprocessing steps that were developed in order to alleviate some of these problems and to render the data more exploitable by the techniques used. Fig. 2.6 illustrates the process. Throughout the rest of the chapter, image 0032 of the sample dataset is used for illustrations for the sake of consistency, except in Section 2.1.2 where image 0007 is used since it displays an illumination gradient that will be clearly visible to the readers.

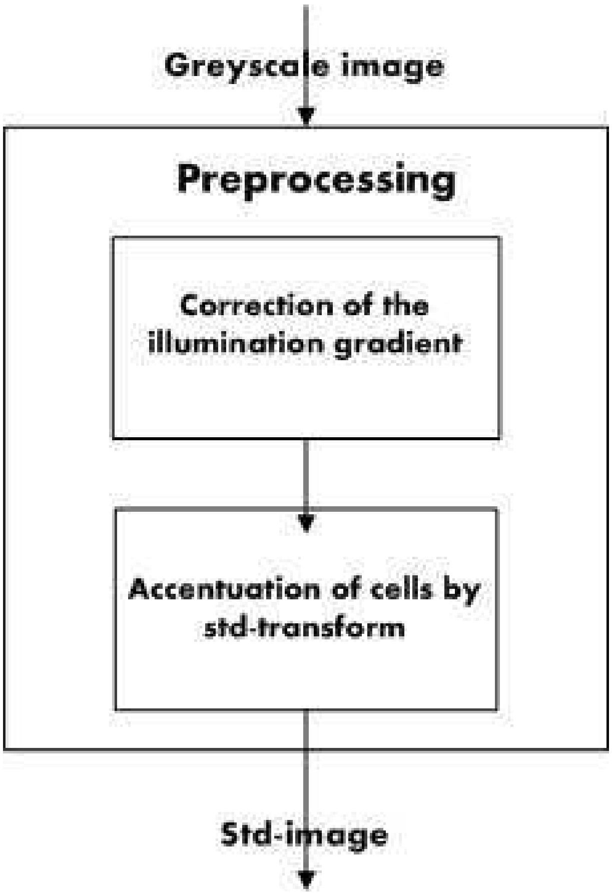


**Figure 2.6:** *Corrective pre-processing schematic.*

### 2.1.2 Removal of the illumination gradient

The problem of uneven field illumination in imagery is a well-known artefact of optics. Berry [14] describes it as a routinely-occurring problem in astronomic images, and Obser [148] offers a solution by observing the difference in greylevel in pixels a certain distance from the axis of symmetry of the image, provided illumination was always from a fixed point in space. Since the direction of the illumination gradient varies across the image set, such a method for adjusting intensity levels is inadequate. Lindblad [121] and Wählby [223] employ a data-driven background correction algorithm to estimate the background with cubic $B$-spline, a simplified, $2D$ relative of which we employ.

The global illumination gradient represents the variation in greylevel not caused by the presence of cells. This variation manifests itself well in the projections of the image along its two axes and is identifiable because of its non-local nature, that is rather than being acute ripples spanning cellular diameters in the projections as in the case of cells, it is a smooth trend in greylevel spanning the entirety of the image.

In the following we will explain the process with reference to only the column component, the row marginal being treated in exactly the same way. Let $\mathbf{I}$ be an image of size $n_1 \times n_2$ then the column marginal drawn for $I$ will respectively be:

$$\mathbf{i}_{.j} = \sum_{i=1}^{n_2} \mathbf{i}(i, j), \; i = 1, \ldots, n_1,$$

where $j$ refers to column indices .The column marginal vector $\mathbf{i}_j = (\mathbf{i}_{.1}, \ldots, \mathbf{i}_{.n_2})^T$ could then be fitted by a polynomial curve $\mathcal{P}$ of the form:

$$P(x) = \sum_{k=0}^{p} b_k x^k, \tag{2.1}$$

Here $p$ is the highest degree of the polynomial used. The choice of this degree of the polynomial is made experimentally, beginning from the linear case upward. The illumination gradient to be modeled saturated the model of degree 3 in the case of some of the images where this gradient was at its most complex. The degree was therefore elevated and chosen at $p = 5$ to cater for any future addition to the image dataset.

The coefficients of this polynomial are the solutions of the system of equations $\mathbf{A}.\mathbf{b} = \mathbf{X}$ where

$$A = \begin{bmatrix} \mathbf{i}_{.1}^0 & \cdots & \mathbf{i}_{.1}^p \\ \vdots & \ddots & \\ \mathbf{i}_{.n_2}^0 & \cdots & \mathbf{i}_{.n_2}^p \end{bmatrix}, \qquad \mathbf{b} = \begin{bmatrix} b^0 \\ \vdots \\ b^p \end{bmatrix}, \qquad \mathbf{X} = \begin{bmatrix} \mathbf{i}_{.1} \\ \vdots \\ \mathbf{i}_{.n_2} \end{bmatrix}, \tag{2.2}$$

where $\mathbf{i}^0_{.1}$ represents the 0th instance of the marginal. As $n_2 \gg p$, the optimal solution is the *pseudo-inverse* solution, that is:

$$\mathbf{b} = [\mathbf{A}^T\mathbf{A}]^{-1}\mathbf{A}^T\mathbf{X}. \tag{2.3}$$

The coefficients of the curves $\mathcal{P}_i$ and $\mathcal{P}_j$ allow us to calculate the approximated marginal vectors $\mathbf{i'}_i$ and $\mathbf{i'}_j$. The corrected image is obtained by subtraction of the value of the two marginals from each pixel value, i.e.

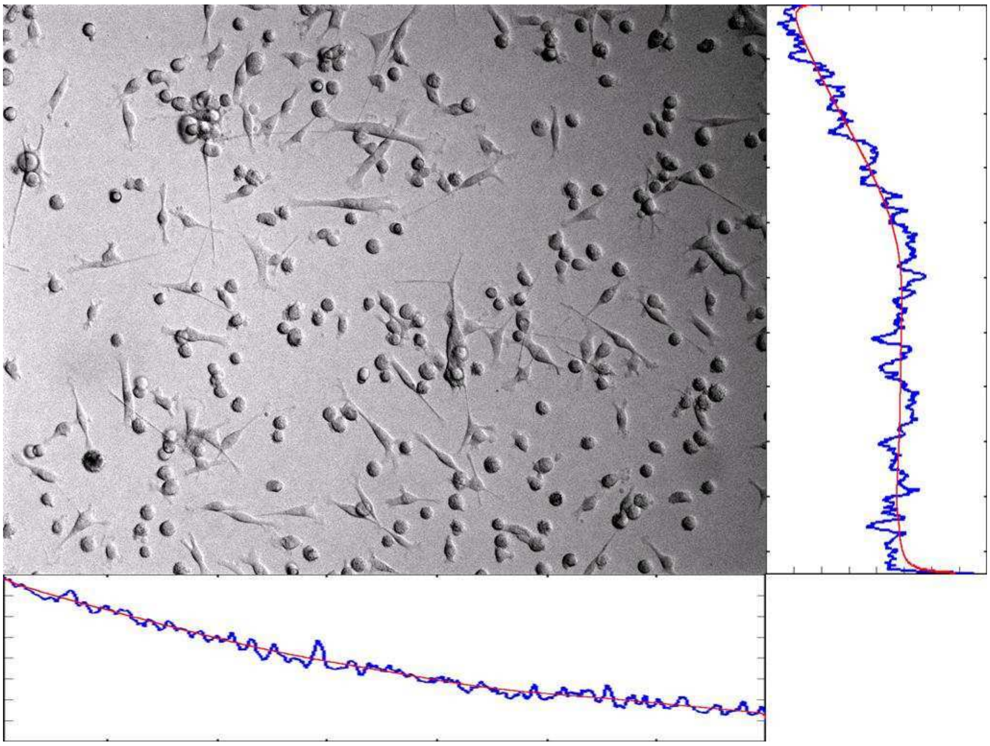
$$\mathbf{I}_{corrected} \leftarrow \mathbf{I} - \mathbf{i'}_{i.} - \mathbf{i'}_{.j} \tag{2.4}$$

Fig. 2.7. depicts the process used to compensate for the illumination gradient. The dotted lines represent the approximated marginals, and their smoothness means that no local noise artefacts are introduced during the correction. This process of modeling the marginals of the image is computationally lighter than a modeling of the image surface, and is adaptive to the image, and does not pre-suppose the direction and the extent of the initial illumination gradient.

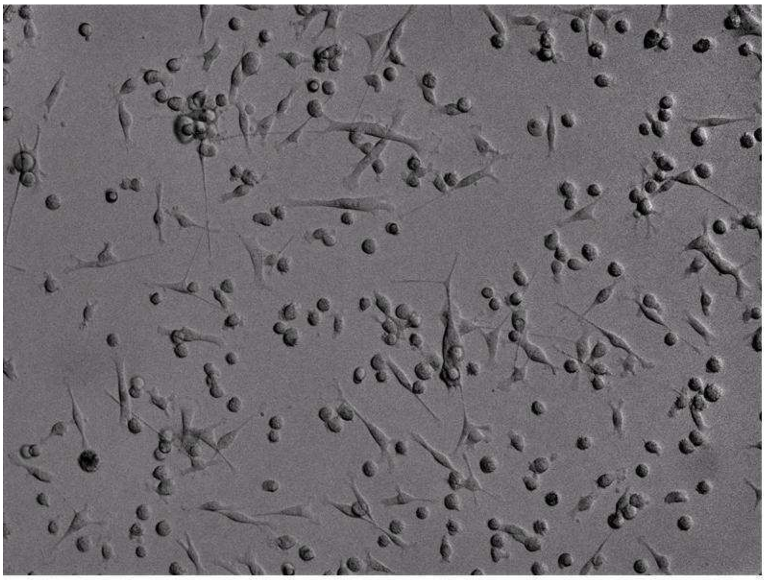
### 2.1.3 Enhancing the cells

The problems of similarity of cell and background intensities, of low-registering spread cells and especially of bright and dark sides of cells mean that the image in this form is unexploitable for segmentation (Fig. 2.10 *d.* in Section 2.2 offers an illustrative example). Cell pixels would be assigned into different classes than in case of histogram based thresholding, and in case of region-bases segmentation the region boundaries will follow these dark-to-light transitions rather than cell boundaries.Therefore greylevel intensities are transformed into a quantity that assumes different values for cells than for the background, and which is invariant regardless of illumination. We know cells are more textured than the background, therefore local standard deviation within cells is greater than without them. Therefore we calculate the local standard deviation image $I'$ in neighbourhoods of $a \times a$ pixels (eq. 2.5) around each pixel of our greylevel image using $a = 3$. This is the smallest support of any spatial filter and is preferred in this case because it makes the differences across the image prominent without adding information from farther away in the image and may be the result of another image feature. The *std*-image can be evaluated as:

$$I'(i,j) = \sqrt{\frac{\sum_{i,j}(I(i,j) - \bar{I})^2}{a^2}}, \qquad \bar{I} = \frac{\sum_{i,j} I(i,j)}{a^2}. \tag{2.5}$$



(a)



(b)

**Figure 2.7:** *a) Plots of the column marginals below and row marginals to the right of an uncorrected image, the plots in red being their approximations; b) image with illumination gradient removed.*

where the pair $(i, j)$ represent the pixel's coordinates. $I'$ has the desireable property of the cells appearing different from the background, thus making them independent of the illumination conditions a particular image has been subjected to.

The image used for this step is the one issued from the illumination-gradient correction algorithm that has been smoothed with a $3 \times 3$ Gaussian filter. This is, once again, the bare minimum smoothing that could be applied, and was therefore selected over filters with a larger support that could add information from neighbouring pixels to the centre pixel.

Fig. 2.10 *b.* shows the standard-deviation image thus produced, juxtaposed to the original in 2.10 *a.* It is discernible from it that cell boundaries and other parts of high local variation register markedly relative to the rest of the image which is darker, and the light and dark parts in the original image both produce similar effect in the std-image.

## 2.2 Image binarisation

At this point, it is appropriate to elucidate some aspects of image segmentation even though we shall review the topic in greater detail in chapters 3 and 4. Image segmentation in our context helps to separate cells from the background and then cells from other cells. However, the first of these goals must be met for us to proceed with cell detection, which mandates its discussion here. This separation of the subject is supported by the logical separation in segmentation concepts utilized as well as the physical separation afforded by the modular nature of the processing chain.

We will variously refer to the segmentation of cells from the background as *image binarisation* because the segmented image is binary i.e. *cell* and $not - cell$. Section 2.2.2 will describe the method used for the binarisation, but the first question that we must contend with is "What image data we need to work on in order to achieve this binarisation?". Of course, the original image has overlapping greylevel ranges for both cells and the background, hence a more contrasted image is required. We have just created such an image i.e. the standard-deviations image in the previous section. That image has much of the central portion of cells at the same greylevel as the background, and binarising it was found to introduce more uncertainty in the process because it left holes at the centre of cells which were computationally indistinguishable from holes between cells in a connected component, and filling them was based on an empirical hole-size estimation. Thus an input image was required which would still maintain the *cell* - *not* − *cell* contrast but would reduce the *cell* - *cell* contrast. Gaussian blurring, which does the second at the cost of the first, guided us to the much better manner of proceeding with the issue, *anisotropic diffusion*. Details of the process follow in the following sections, and an overview in Fig. 2.8.

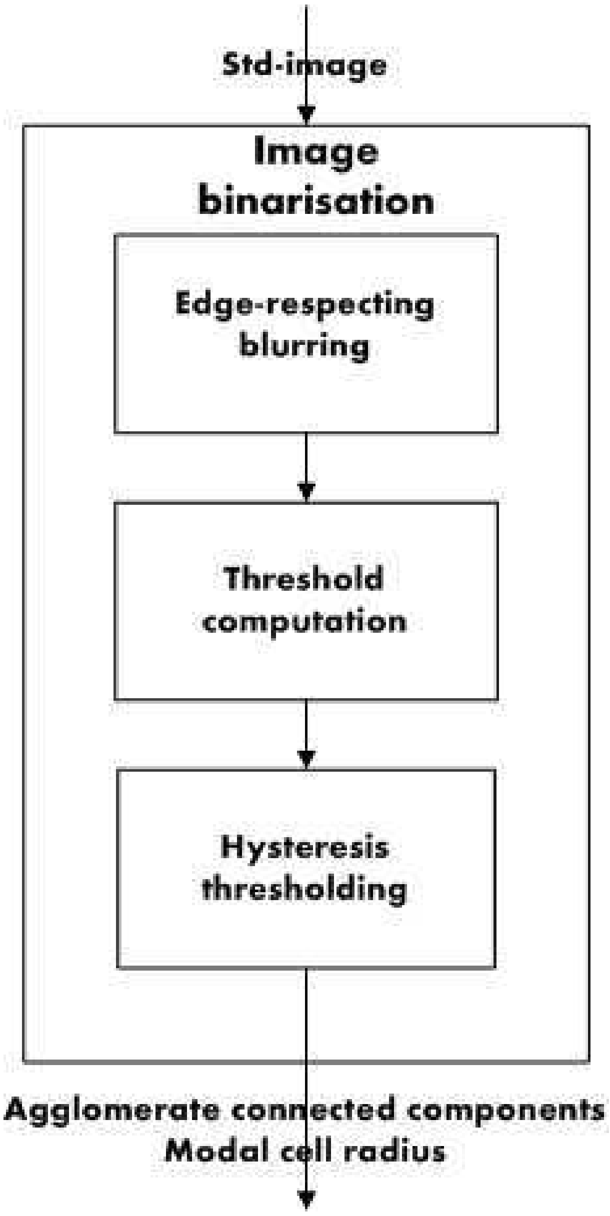


**Figure 2.8:** *Image binarisation schematic.*

### 2.2.1 Calculating the image to binarise: anisotropic diffusion

Since the elegant formulation introduced by Perona and Malik [160], anisotropic diffusion has been an important image enhancement technique aiming at reducing image noise without removing significant parts of the image content, typically edges, lines or other details that are relevant for the interpretation of the image. Anisotropic diffusion resembles the process that creates a scale-space, where an image is at the origin of a parameterized family of successively more and more blurred images based on a diffusion process. Each of the resulting images in this family are given as a convolution between the image and a $2D$ isotropic Gaussian filter, where the width of the filter increases with the parameter. This diffusion process is a linear and space-invariant transformation of the original image. Anisotropic diffusion is a generalization of this diffusion process: it produces a family of parameterized images, but each resulting image is a combination between the original image and a filter that depends on the local content of the original image. As a consequence, anisotropic diffusion is a non-linear and space-variant transformation of the original image.

Formally, let $\Omega \subset \mathbb{R}^2$ denote a subset of the plane and $I(\cdot, t) : \Omega \rightarrow \mathbb{R}$ be a family of greylevel images, then anisotropic diffusion is defined as:

$$\frac{\partial I}{\partial t} = \operatorname{div}\left(c(x,y,t)\nabla I\right) = \nabla c \cdot \nabla I + c(x,y,t)\Delta I \tag{2.6}$$

where $\Delta$ denotes the Laplacian, $\nabla$ denotes the gradient, div(.) is the divergence operator and $c(x, y, t)$ is the diffusion coefficient. $c(x, y, t)$ controls the rate of diffusion and is usually chosen as a function of the image gradient so as to preserve edges in the image. Perona and Malik proposed two functions for the diffusion coefficient:

$$c\left(||\nabla I||\right) = e^{-(||\nabla I||/K)^2} \tag{2.7}$$

and

$$c\left(||\nabla I||\right) = \frac{1}{1 + \left(\frac{||\nabla I||}{K}\right)^2} \tag{2.8}$$

the constant $K$ controls the sensitivity to edges and is usually chosen experimentally or as a function of the noise in the image.

Anisotropic diffusion can be used to remove noise from digital images without blurring edges. With a constant diffusion coefficient, the anisotropic diffusion equations reduce to the heat equation which is equivalent to Gaussian blurring. This is ideal for removing noise but also indiscriminately blurs edges too. When the diffusion coefficient is chosen as an edge seeking function, such as in Perona and Malik, the resulting equations encourage diffusion (hence smoothing) within regions and prohibit it across strong edges. Hence the edges can be preserved while removing noise from the image.

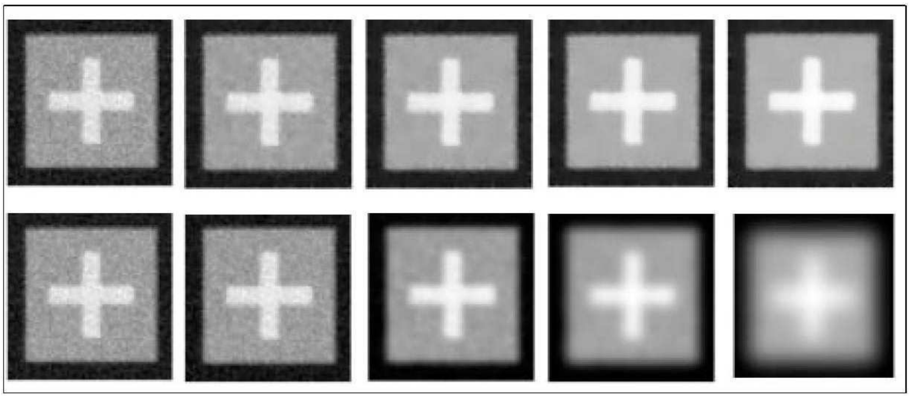


**Figure 2.9:** *Comparison on a toy problem between Gaussian filtering (bottom) and Anisotropic diffusion (top) at increasing scales.*

We use anisotropic diffusion as an edge-preserving blurring operator, with Perona and Malik's first stopping criteria is proportional to $||\nabla I||$ and hence higher-gradient parts of the image are the least to diffuse.

### 2.2.2 Selecting a thresholding: Otsu's criterion

Otsu's method [152] belongs to the class of clustering-based thresholding algorithms, the greylevel data undergoes a clustering analysis, with the number of clusters being set always to two. Otsu suggested minimizing the weighted sum of within-class variances of the foreground and background pixels to establish an optimum threshold. Recall that minimization of withinclass variances is tantamount to the maximization of between-class scatter. This method gives satisfactory results when the numbers of pixels in each class are close to each other. The Otsu method still remains one of the most referenced thresholding methods.

The way of accomplishing this is to set the threshold so as to try to make each cluster as tight as possible, thus minimizing their overlap. Obviously, we can't change the distributions, but we can adjust where we separate them (threshold them). As we adjust the threshold one way, we increase the spread of one and decrease the spread of the other. The goal then is to select the threshold that minimizes the combined spread. The algorithm operates directly on the grey level histogram [e.g. 256 numbers, $p(i)$], so it is fast (once the histogram is computed).

We can define the within-class variance as the weighted sum of the variances of each cluster:

$$\sigma^2_{within} = \frac{n_B(T)\sigma^2_B(T) + n_O(T)\sigma^2_O(T)}{n} \tag{2.9}$$

where

$$n_B(T) = \Sigma_{i=0}^{T-1} p(i), \quad n_O(T) = \Sigma_{i=T}^{N-1} p(i), \quad n = n_B + n_O, \tag{2.10}$$

$$\sigma^2_B(T) = \textit{the variance of the background pixels (below threshold)}, \tag{2.11}$$

$$\sigma^2_O(T) = \textit{the variance of the foreground pixels (above threshold)}, \tag{2.12}$$

and $[0, N-1]$ is the range of intensity levels.

Otsu's method assumes the histogram of the greylevel image to be bimodal; has no implication of spatial coherence, nor any other notion of object structure; and assumes uniform illumination (implicitly), so the bimodal brightness behavior arises from object appearance differences.

Once the anisotropically diffused image has been created, an Otsu threshold is obtained using the algorithm mentioned above, and the diffused image is then segmented using a hysteresis procedure with the higher threshold equal to the Otsu threshold and a lower one equal to 40% of it as is often the practice (see [227]).

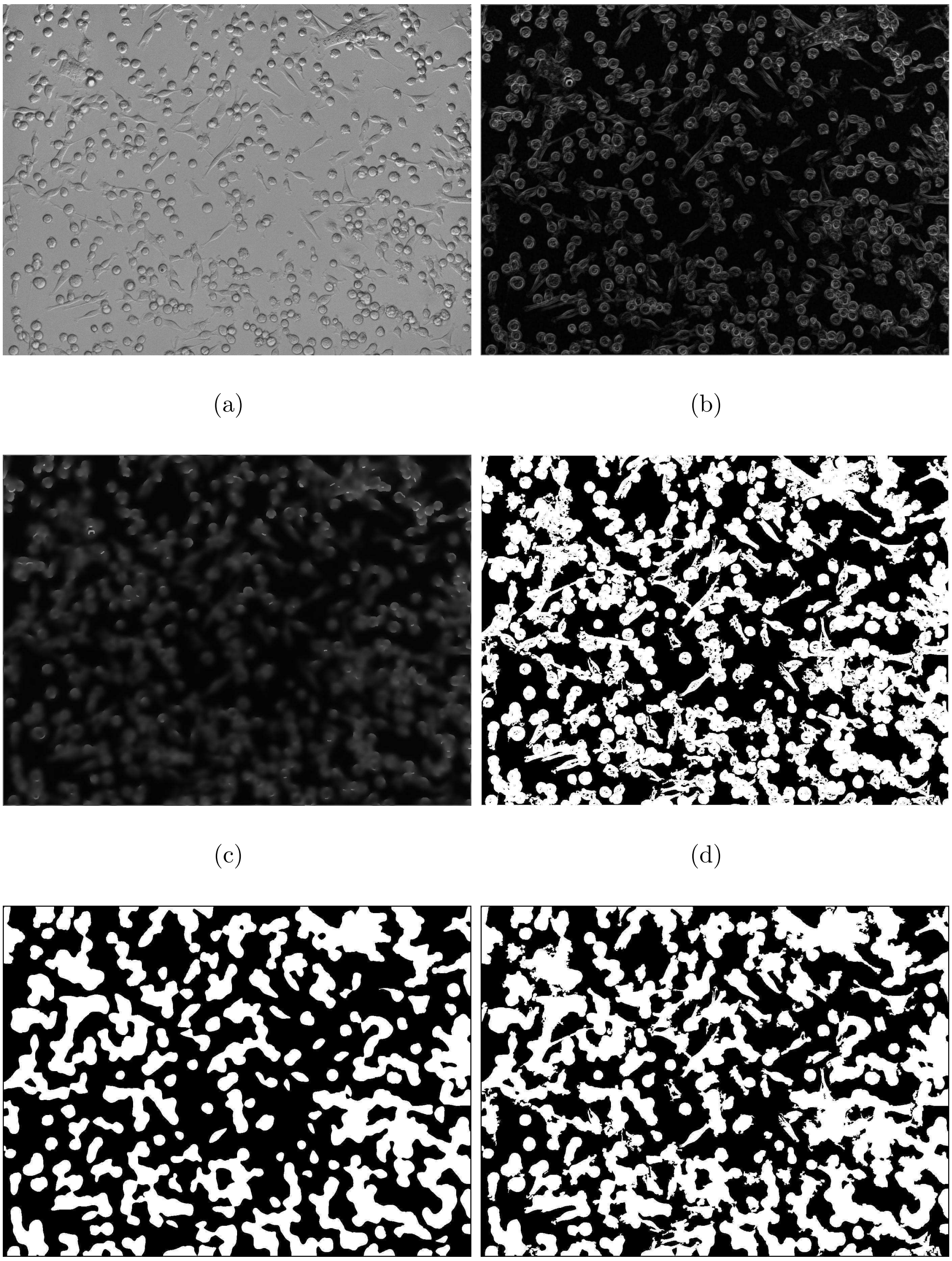
### 2.2.3 Thresholding the image: hysteresis

Now that we have seen how to obtain the value of a threshold that best separates the $cell$ and $not - cell$ pixels according to Otsu, we proceed to applying this threshold in the manner that gives us the best separations between cells and backgroud in the image.

An important feature of the image is that although many cell pixels are darker than some of the more prominent ones near cell walls, these darker pixels are nevertheless usually connected to other lighter ones through cell texture continuity up to the cell wall and within-cell ridges. Therefore, if we can identify the most prominent pixels, from them we can also obtain the less significant pixels connected to them, while avoiding background pixels of comparable intensities to bleed-through into the class *cell*. This is essentially the principle behind *hysteresis thresholding* used in Canny's edge detector [30] for small ridge suppression.

If $I$ is our grayscale input image, then given intensity thresholds $t_h$, $t_l$, we can construct two sets of pixels, $H = \{(x, y)|I(x, y) = t_h\}$ and $L = \{(x, y)|I(x, y) = t_l\}$, which can be used directly to produce binary masks such that all pixels above the threshold are "on" and all others are off. For a given image, $t_h$ marks the level at which background pixels are negligible, while $t_l$ indicates the point below which no more cell pixels are expected. Both these values can be either global, $t$, or locally adaptive, $t(x, y)$. Empirically, the use of Otsu's threshold for $t_h$ and its fraction for $t_l$ tends to produce good results. With these two sets, we now wish to obtain $C$, such that $H \subset C \subset L$. In short, $H$ is too restrictive and removes too many cell pixels, while $L$ is too permissive and keeps too many background ones. This implies that our target set $C$ consists of $C = H \cup L'$, where $L' \subset L$. Now, all we need is a heuristic or criterion for choosing $L'$ out of $L$. This is supplied by a connectivity criterion - a pixel in $L$ having in its 8-neighbourhood a pixel in $H$ is included in $L'$.

In the detection of $1D$ features, the widespread acceptance of Canny's edge-detection algorithm is due in large part to its use of thresholding with hysteresis. We adapt it to detect $2D$ features i.e. connected components thus: 1. Select a starting pixel whose greylevel (note that we are working on an anisotropically diffused image) above an upper threshold $t_h$. Mark that pixel as having been visited. 2. Select and move to an adjacent pixel whose greylevel is above the lower threshold $t_h$. Mark that pixel as having been visited. 3. Repeat 2 until the value of the greylevel at the selected pixel falls below a lower threshold $t_l$. 4. Repeat 1 until all pixels above $t_h$ have been marked as visited.



(a) (b)

(c) (d)

(e) (f)

**Figure 2.10:** *Image* 0032 *(a) original corrected for the illumination gradient, (b) standard-deviations image, (c) anisotropically diffused image, (d) binarisation of (b), (e) binarisation of (c), (f) final binarisation: sum of (d) and (e). Notice the absence of holes inside of cells in (e).*

Figure 2.10 sums up the image binarisation process and at the same time describes its dependence on the earlier pre-processing steps. We begin from the original image, implying the illumination-gradient correction, pass through to the standard-deviation image, and then to the binarisation step based on thresholds calculated on the aforementioned images.

This pre-processing is a pre-requisite to the detection of cells that we present in the following sections in that cell detection works on the principle of detecting the image features of the standard-deviation, and uses the binarised image, or as we like to refer to it as the image of *agglomerated connected components* (*ACC*) to avoid cell detection in the background. Of course, these images prove useful once again when image segmentation is performed.

## 2.3 Cell detection

Cell detection, by which we understand cell recognition and counting, in microscopic systems is an interesting challenge due to the nature of cells i.e. their shape and adhesion qualities, the presence of debris and of noise, and the difficulties linked to the culture and imaging techniques (staining versus none, fixing, lighting and other apparatus-related issues). To evaluate cell distribution, types and migration it is required to count either the total number of cells or that of the cells which have migrated during end-point analysis. A correct identification of these objects usually requires a combination of the segmentation and post-processing stages described above, but sometimes a bespoke object recognition and counting procedure e.g. when a count is needed before the application of those stages or is necessitated by the quality of the image or of the cells.

The counting of cells is a frequently-encountered task in microscopy imagery. A cell count as a byproduct of segmentation, as mentioned before, is usually the obvious method of choice. This indirect approach to the problem of counting cells could view segmented objects as individual cells [232] or use globally estimated features as intensity of density or density of colour to approximate the cell quantity [164] [130].

Faustino et al [60] also use histogram-thresholding on fluorescent cellular blobs with the calculation of the resulting centroids, and present an interesting comparison with manual counting performed by a panel of experts, showing the significant diversity in manual counts and the difficulty of validating automatic counts. We have presented a similar study that could be found in [211].

Sio et al.[188] addressed the problem of parasitemia estimation using edge detection and splitting of large clumps made up from erythrocytes. The outcome of the approach was shown to be satisfactory for well-stained samples with well-separated erythrocytes. For the same problem, watershed transform [215] was also employed, given that local maxima indicate the centers of convex shapes, i.e. blood components particularly erythrocytes. This concept, however, is only justifiable for images which exhibit a small degree of cell overlap.

Multi-scale blob detection [125] appears to be have cell detection as an obvious application but is not very frequently applied to real cell images, however it has been applied in some cases with success e.g. by [159] who then use a multi-scale active contour to segment those cells.

Recently a *pixel patch* strategy has been formulated as described in [146]. The process involves two stages: preprocessing and classification. The major task of preprocessing is to derive a representation of cells which makes subsequent classification computationally effective and insensitive to environmental changes by providing the classifier only with information essential for recognition. In the classification stage, a neural network is trained to determine if a pixel patch contains a centered cell body. This is done with pixel patches represented by feature vectors derived in preprocessing. Although Long et al [123] simplify it by not learning the not-cell patch, and following the calculation of the confidence map by a local maxima detection to designate cell centres, higher accuracy rotation-invariance are obtained by Theis et al [200] [201] using unsupervised independent component analysis with correlation comparison. In order to account for a larger variety of cell shapes, they also propose a directional normalization.

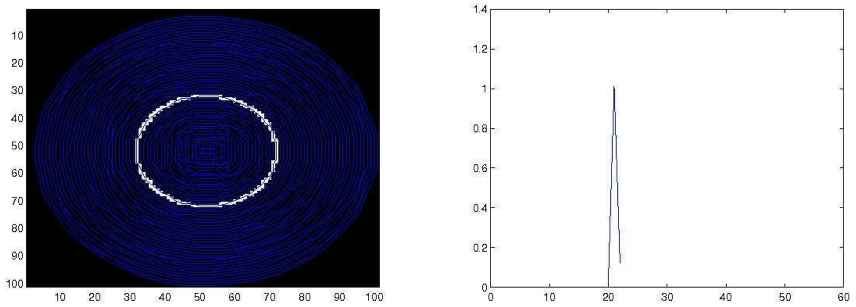
Matched filtering [207] could be used to construct such a search space since it maximizes the signal-to-noise ratio between a template and parts of the target image, thereby producing a response space similar to Hough with peaks corresponding to regions of the image that are most similar to the template. Locating these peaks amounts to the aforementioned cell detection.

A template matching approach, also exploiting selective-marking, is used by Kachouie et al [96] to separate live and dead cells, stained in different colours, and then correlation maps are developed for each by template matching with a cell-sized disc, the maxima indicating cell centres. Template-matching has also seen to work well in case of images of other objects, e.g. by Popescu et al. [163] to detect and count tree crowns.

The particular nature of our images, as described in section 2.1 dictates our choice of method. The data does not lend itself to correct segmentation into connected components each containing potentially one cell. Therefore, direct cell segmentation is likely to produce incoherent results and has to be preceded and therefore guided by this in-place cell detection. A similar approach has also been taken by Pinzón et al. [162] suggested that the problem of erythrocyte segmentation could be reduced to peak selection in the Hough circle space. The study focused on detecting erythrocytes of circular shape and uniform size, an assumption which has to be relaxed for the purpose of our study.

### 2.3.1 The "Halo" filter

The problem of locating cells can be reformulated into a peak-finding problem in a space of correlation with a matched filter. We notice that in the image of local standard deviations (Fig. 2.10 *b.*) cells are represented by closed rings where the cell walls would be, enclosing a textured interior. Most of the time these rings are quasi-circular, corresponding to the cell walls of round cells and the central, thicket portion of spread cells. Correlation with a matched filter should maximize the signal-to-noise ratio where these structures are present.



(a) (b)

**Figure 2.11:** *Correlation coefficient (b) between a circlular object and circles of varying radii (a) peaks at* 20*, the radius of the drawn object*

The idea can be more easily understood on a binary image, where objects being represented by a value equal to 1, a circular object will give a higher coefficient of correlation with a concentric circle of the same size (Fig. 2.11). Indeed the correlation would be 1 if the object is a perfect circle. Therefore if $X$ is a vector of size $n$ of intensity values selected in a circular fashion from a greylevel image containing a circle on a background at 0, and $Y$ be our filter with 1s arranged in a circular manner in 2-dimensions, then the Spearman's rank correlation coefficient [191]:

$$\rho = \frac{n\sum_i X_i Y_i - \sum_i X_i \sum_i Y_i}{\sqrt{n\sum_i X_i^2 - (\sum_i X_i)^2}\sqrt{n\sum_i Y_i^2 - (\sum_i Y_i)^2}} \tag{2.13}$$

reduces to:

$$\rho = \frac{\sqrt{n-1}\sum_i X_i}{\sqrt{\sum_i X_i^2 - (\sum_i X_i)^2}} \tag{2.14}$$

and therefore the maximum value of $\rho$ corresponds to the radius of the filter that best matches that of the object.

We develop on this idea of correlation, used by both template matching and matched filtering, without actually matching patterns with respect to correlation or of the signal-to-noise-ratio, in the sense of template matching and matched filtering respectively. The idea of correlation is used to construct a filter, rather like a matched filter, in a manner adapted to the size of cells, as we shall see in the following.

We propose a ring-shaped matched filter constructed conditionally from two $2D$ Gaussian functions $G_1(\vec{x}) \sim N(\mu_1, \sigma_1)$ and $G_2(\vec{x}) \sim N(\mu_2, \Sigma_2)$ and has the equation:

$$f(\mathbf{x}) = \begin{cases} \frac{1}{2\pi\Sigma_1} \times \exp^{-\frac{(\mathbf{x}-\boldsymbol{\mu}_1)^T \Sigma_1^{-1}(\mathbf{x}-\boldsymbol{\mu}_1)}{2}} & \text{if } G_1(\mathbf{x}) < G_2(\mathbf{x}) \\ 0 & \text{otherwise} \end{cases} \tag{2.15}$$

where $\vec{x} = (x_1, x_2)^T$ defines pixel location, $^T$ denotes the transpose operator, $(\mu_i, \Sigma_i)_{i=1,2}$ the Gaussian parameters. The supposition is that $\Sigma_i$ is a diagonal matrix of variances i.e. the Gaussian function is circular. An equilibrium between the radii was empirically selected at $\Sigma_2 \approx 2\Sigma_1$ in order to match the shape of the cellular rings defined by the form of round cells and the central part of spread cells.

Fig. 2.12 provides a summary of the detection process that follows.

### 2.3.2 Auto-calibration of the Halo filter support

For each image, the filter must adapt its size to the size of cells in the images that varies with the zoom factor, the cellular line used and natural variation among a cell population. To make the filter autoadaptive to the modal radius of the cells in a certain image, we use radius histogramming [91] to determine the modal cell radius in the following way. The determination of the representative cell radius in an image implies the determination of the value of $\Sigma_2$ as described above, the two quantities being equal.

Before this determination could begin, the image is cleaned of connected components too large to consist of one or two cells, so the modal radius is indeed close

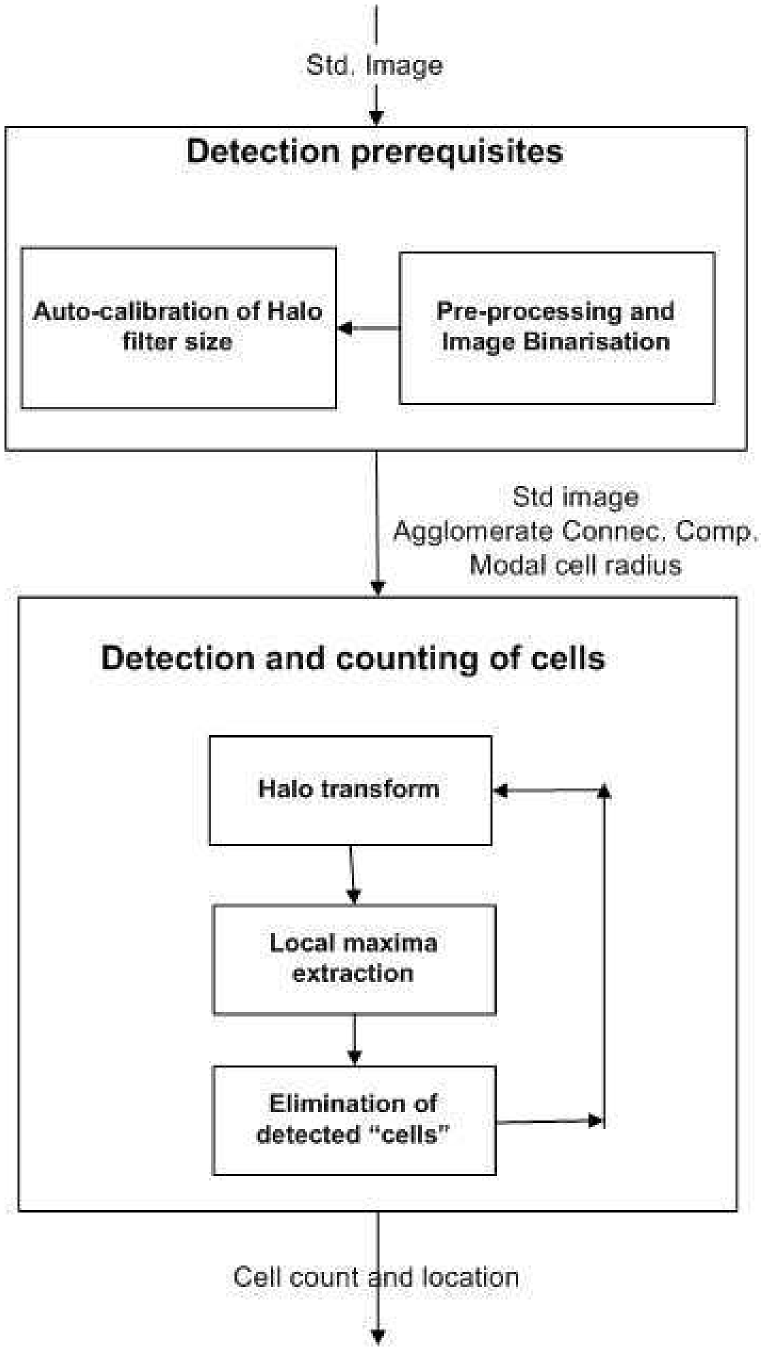
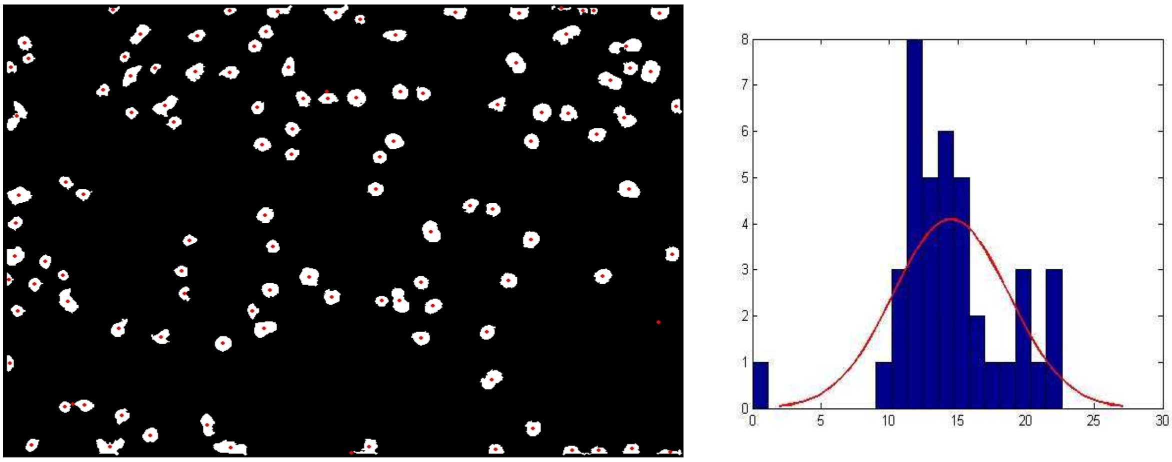


**Figure 2.12:** *Image binarisation schematic.*

to the true average radius. This is performed through a morphological opening by a circular structuring element that was comfortably larger than two cells put side by side, the average size of a cell having been empirically sampled over several images.

With the knowledge of the position of the centroid of a connected component (Fig. 2.13 *a*.) and of the contour points, we calculate for each connected component, the distance between the points of its contour and its centroid (Fig. 2.14). Then a histogram of these distances is calculated. The representative radius of all the connected components in the image is chosen to be the modal value i.e. the peak of the histogram $p$, *cf*. figure 2.13 *b*.



(a) (b)

**Figure 2.13:** *a) Connected component boundaries are known and centroids could be calculated. b) The radius histogram with a mode of* 16 *pixels.*

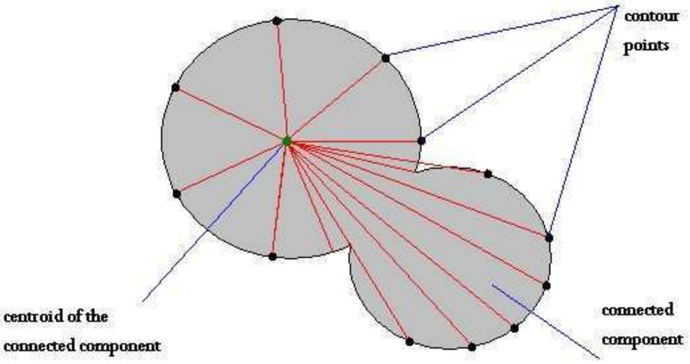


**Figure 2.14:** *Distances from connected components' borders to their centroids are calculated.*

### 2.3.3 The "Halo" transform and localization of peaks

Convolutions $\mathbf{f}*I$ between the "halo" filter $\mathbf{f}$ of size $p \times p$ are calculated with the standard-deviations image $I$. The filtered image has the characteristic "halos" representing cell portions- hence the name *halos image* which we will use henceforth. For relatively round cells this halo has a bright umbra where correlation scores are highest. In case of agglomerates of cells, halo umbras may join together to form ridges, as are also formed in the case of spread cells. The pixel where the coefficient of correlation was maximum is then located being the only point in the intersection of halos image function and its greyscale dilation, since dilation makes plateaux around peaks of the same height as the peak and the shape of the structuring element (square having a side equal to the modal radius in this case). Fig. 2.16 illustrates this principle. This *peak* implies the presence of a cell, and is henceforth referred to as the cell's *centre*.

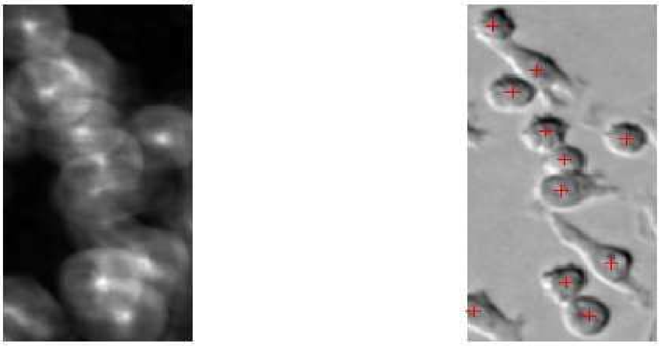
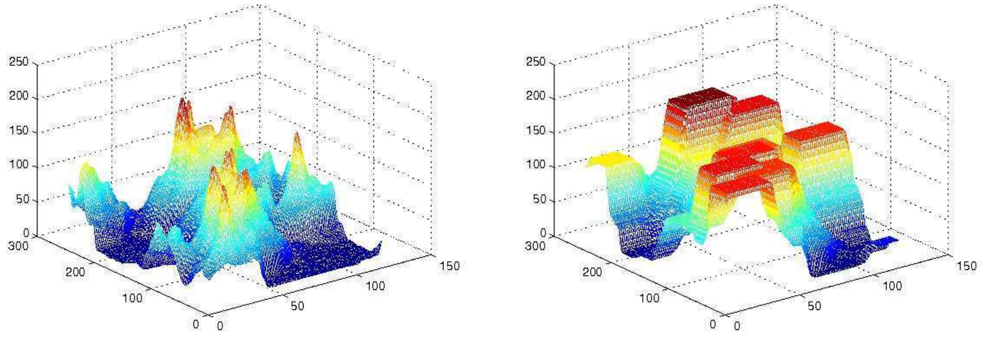

(a) (b)

**Figure 2.15:** *A zoom on a halos image and the peaks in the correlation space superimposed on the original image.*



(a) (b)

**Figure 2.16:** *3D plot of (a) halos, (b) greylevel dilation around each peak.*

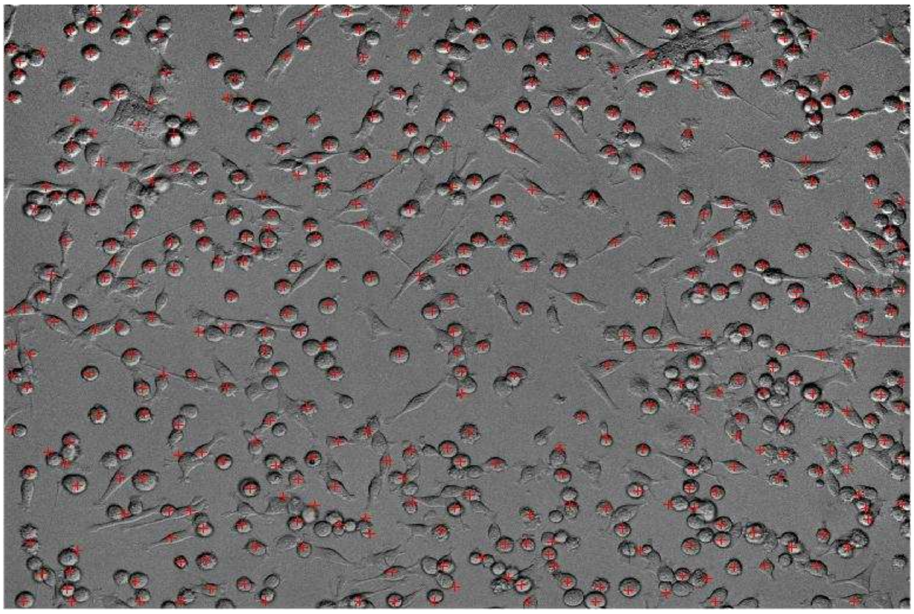


**Figure 2.17:** *Image* 0032 *with cell centres superimposed on the original image.*

Since the counting of cells in a given image is a key element in the biologists' analysis which requires the proportion of each type of cell for arriving at its decision concerning the metastasic potential of this cell sample, we have decided to introduce a validation step for cell centres thus detected, in order to reject the centres produced due to complex interaction among agglomerated cells, the windowing nature of the filter and image border effects, and the statistical bias of the modal radius parameter that then dictates peak detection. This step is described in the following section.

## 2.4 Cell validation by a maximum likelihood test

Following Yves Meyer's seminal work [141], signal decomposition models into a geometrical component and a textured component have recently been proposed in image processing and has proved useful in signal decomposition [192], texture modeling [209], image denoising and restoration [6] [151]. The classical approach consists in considering that an image $f$ can be decomposed into two components $u + v$. The first component $u$ is well-structured, and has a simple geometric description: it models the homogeneous objects which are present in the image. The second component $v$ contains both textures and noise.

The filtering scheme described in section 2.3 identifies cells but can produces errors such as provoking peaks in between cells clustered together, or causing two peaks on spread cells. We define a centre's quality of detection as the degree of confidence with which we can state that it is contained within a cell. This amounts to saying that the said cellular connected component is indeed a cell. This is a classical problem of the detection of a signal $u$ in the presence of noise $v$, and can be formulated into a maximum of likelihood test as we describe shortly.

The decomposition theory stipulates that $u$ and $v$ are additive components, but we wanted to make sure if it indeed was the case. Therefore, we try to ascertain the nature of the noise component.

### 2.4.1 Determining the nature of the noise

The identification of the nature of the noise affecting the image is an important step in any system of interpretation of information by vision when the nature of the degradation is unknown. In an initial study K. Chehdi and Mr. Sabri [36] have shown that it is possible to identify the nature of the noise from the recording of changes in statistical local (standard deviation based on the average) calculated in homogeneous regions of observation alone. The decision criterion uses these statistics to local identify the nature of the noise.

The characterization of the presence of additive noise or multiplicative in an image based on the following hypothesis: *In homogeneous zones of the degraded image, only noise statistics are involved.*

Based on this assumption, it was shown in [9] that the standard deviation $\sigma[f_h]$ of a homogeneous part $f_h$ of the image $f$ can be written as:

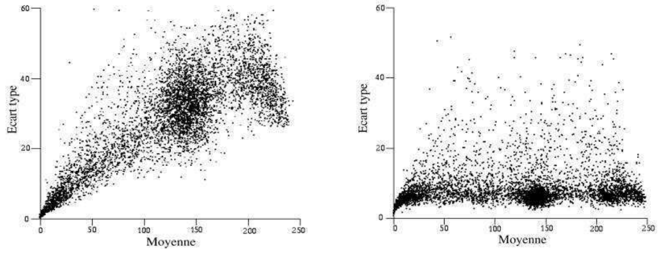
$$\sigma[f_h] \approx \sigma[b] \quad \textit{in case of additive noise} \tag{2.16}$$

where $b$ is an additive white noise, centered and non-correlated with the image observed $f$.

$$\sigma[f_h] \approx [f_h] \times \sigma[n] \quad \textit{in case of multiplicative noise} \tag{2.17}$$

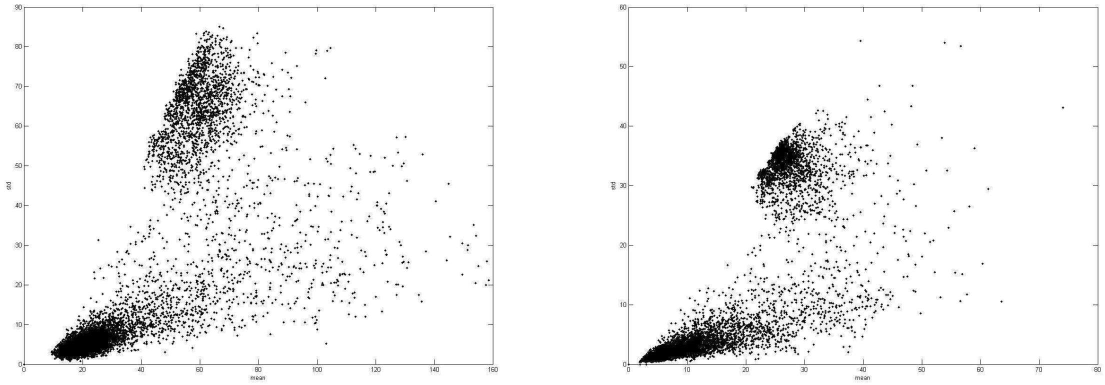
where $n$ is a multiplicative white noise, with a mean equal to unity and non-correlated with the image $f$.

In other words, according to equations 2.16 and 2.17, if the variation of local standard deviations forms a line *parallel to the local means, the noise is multiplicative.* If this variation *follows a straight line through* 0*, the noise is additive.* This is illustrated by the graphs in Fig. 2.18, obtained by an artificial degradation by, respectively, a centered additive noise of standard deviation 10 and a multiplicative noise of unit mean of standard deviation 0.2.



(a) (b)

**Figure 2.18:** *Evolution of the standard deviation based on the average; (a) case of an image degraded by additive noise, (b) case of an image degraded by multiplicative noise. Figure courtesy [9].*



(a) (b)

**Figure 2.19:** *Examples of the noise evolution plots for a couple of the dataset images.*

The noise evolution plots represented in Fig. 2.19 are representative of those obtained for the entire dataset. They clearly show the pattern of an additive noise, i.e., is scattered in a funnel around a straight line passing through the origin, and hence this is the nature of noise we use to formulate our signal-and-noise model below:

### 2.4.2 The decision theory

Formally, an observation $I$ could be modeled by adding an independent zero-mean, unit-variance Gaussian random variable to each component of the target signature represented by the matched "Halo" filter. That is

$$I = S + B, \tag{2.18}$$

where $S$ is the template (i.e. the known signal) image and $B$ is the background clutter noise. Let the columns of pixels $(I_{\cdot 1}, I_{\cdot 2}, \ldots, I_{\cdot n})$ that compose the observed image be random vectors in sequence. In a similar manner, $S = (S_{\cdot 1}| \ldots |S_{\cdot n})$ and $B = (B_{\cdot 1}| \ldots |B_{\cdot n})$ are $m \times n$ matrices and $B_{\cdot i}$ and $S_{\cdot i}$ are $m$-dimensional vectors. The $I_{\cdot i} = (I_{1i}, \ldots, I_{mi})^T, i = 1, \ldots, n$ are assumed to be drawn from the same distribution and thus to be independent and identically distributed.

The cell detection problem can therefore be modeled as the hypothesis testing problem of a null hypothesis $H_0$, where $S_{\cdot i} = (0, 0, \ldots, 0)^T$, against an alternate hypothesis $H_1$, where $S_{\cdot i} = (S_{1i}, \ldots, S_{mi})^T$. That is:

$$\begin{cases} H_0 : & I = B \\ H_1 : & I = S + B \end{cases} \tag{2.19}$$

Two assumptions are made on the noise probability distributions, that the column components of noise are independent, and that their distributions are Gaussian. The independence assumption is solely a practical requirement for the purpose of solving the equations involved. The gaussianity assumption comes from signal-space analysis of considering $n$-dimensional pictures of image points. Usually when a binary decision should be taken from a multiple observations vector [221], the choice of the Gaussian cumulative density function is more interesting when computing the likelihood ratio and lead to interesting graphical interpretation. Hence the noise components are independent, and the density of $B$ is the product of $n$ Gaussian densities. Therefore the likelihood ratio is:

$$\Lambda(I) = \frac{p(I_{\cdot 1}, I_{\cdot 1}, \ldots, I_{\cdot n}|H_1)}{p(I_{\cdot 1}, I_{\cdot 1}, \ldots, I_{\cdot n}|H_0)} \tag{2.20}$$

$$= \prod_{i=1}^{n} \frac{p(I_{\cdot i}|H_1)}{p(I_{\cdot i}|H_0)} \tag{2.21}$$

$$= \prod_{i=1}^{n} \frac{\frac{1}{(\sqrt{2\pi})^m |\Sigma_{\cdot i}|^{-1/2}} \exp -\frac{1}{2}(I_{\cdot i} - S_{\cdot i})^T \Sigma_{\cdot i}^{-1} (I_{\cdot i} - S_{\cdot i})}{\frac{1}{(\sqrt{2\pi})^m |\Sigma_{\cdot i}|^{-1/2}} \exp -\frac{1}{2} I_{\cdot i}^T \Sigma_{\cdot i}^{-1} I_{\cdot i}} \tag{2.22}$$

where $\Sigma_{\cdot i}$ is the actual noise covariance matrix.

Equation (2.22) can be simplified by taking the logarithm and combining terms:

$$\log \Lambda(I) = \sum_{i}^{n} S_{\cdot i}^T \Sigma_{\cdot i}^{-1} I_{\cdot i} - \frac{1}{2} \sum_{i}^{n} S_{\cdot i}^T \Sigma_{\cdot i}^{-1} S_{\cdot i} \tag{2.23}$$

Finally, if the second term is combined with the logarithm of the original threshold $\Lambda$, the decision rule can be stated as a threshold test on the weighted sum of the $I_{\cdot i}$:

$$\ell(I) = \sum_{i}^{n} S_{\cdot i}^T \Sigma_{\cdot i}^{-1} I_{\cdot i} \underset{d_1}{\overset{d_0}{\lessgtr}} \log \lambda + \frac{1}{2} \sum_{i}^{n} S_{\cdot i}^T \Sigma_{\cdot i}^{-1} S_{\cdot i} = \lambda' \tag{2.24}$$

The statistic that has been identified as $\ell(I)$ is obviously a sufficient statistic for this problem: it will tell in which decision region $I$ lies. Suppose that we can define by $I_1 = \{I|\ell(I) < \lambda'\}$ and $I_2 = \{I|\ell(I) > \lambda'\}$ two subsets of the $n$-dimensionnal space. Therefore the conditional probability that $I \in I_1$ (resp. $I_2$) is just the conditional probability that $\ell(I)$ is less (resp. greater) than $\lambda'$. The error probabilities are therefore:

$$P(d_1|H_0) = \int_{\lambda'}^{+\infty} p(\ell|H_0) d\ell \tag{2.25}$$

$$P(d_0|H_1) = \int_{-\infty}^{\lambda'} p(\ell|H_1) d\ell \tag{2.26}$$

Since $\ell(I)$ is a weighted sum of Gaussian variables, it is a simple Gaussian variable whose variance is constant and whose mean depends on the image. Hence to find the densities $p(\ell|H_0)$ and $p(\ell|H_1)$, we need only find the means and the variance of $\ell(I)$. The conditional means are the expected values of their respective distributions:

$$E[\ell(I)|H_0] = E\left[\sum_{i}^{n} S_{\cdot i}^T \Sigma_{\cdot i}^{-1} B_{\cdot i}\right] = 0 \tag{2.27}$$

$$E[\ell(I)|H_1] = E\left[\sum_{i}^{n} S_{\cdot i}^T \Sigma_{\cdot i}^{-1} (S_{\cdot i} + B_{\cdot})\right] \tag{2.28}$$

$$= \sum_{i}^{n} S_{\cdot i} \Sigma_{\cdot i}^{-1} S_{\cdot i} = \|S\|_{\Sigma}^2 \tag{2.29}$$

The variance of $\ell(I)$ will be

$$\operatorname{var}(\ell(I)|H_k) = \sum_{i=1}^{n} S_{\cdot i}^T \Sigma_{\cdot i}^{-1} S_{\cdot i} = \|S\|_\Sigma^2, \qquad k = 1, 2 \tag{2.30}$$

The probability density $p(\ell|H_0)$ is therefore given by:

$$p(\ell|H_0) = \frac{1}{\sqrt{2\pi}\|S\|_\Sigma^2} \exp\left(\frac{-\ell^2}{2\|S\|_\Sigma^2}\right) \tag{2.31}$$

so that $p(d_1|H_0)$ becomes

$$p(d_1|H_0) = \int_{\lambda'}^{\infty} \frac{1}{\sqrt{2\pi}\|S\|_\Sigma^2} \exp\left(\frac{-\ell^2}{2\|S\|_\Sigma^2}\right) d\ell \tag{2.32}$$

If we substitute $\eta = \frac{\ell}{\|S\|_\Sigma}$, then we have:

$$p(d_1|H_0) = \int_{\frac{\lambda'}{\|S\|_\Sigma}}^{\infty} \frac{1}{\sqrt{2\pi}} \exp(\frac{-\eta^2}{2}) d\eta = \Phi(\frac{\lambda'}{\|S\|_\Sigma}), \tag{2.33}$$

where $\Phi(\cdot)$ is the cumulative distribution function of the zero-mean, unit-variance normal distribution. Since from (2.24), $\lambda'$ is given by $\lambda' = \log\lambda + \frac{1}{2}\|S\|_\Sigma^2$, the probability of false alarm becomes:

$$p(d_1|H_0) = \Phi\left(\frac{\log\lambda}{\|S\|_\Sigma} + \frac{\|S\|_\Sigma}{2}\right) \tag{2.34}$$

Defining $\delta = \|S\|_\Sigma$, then we have the familiar form: $p(d_1|H_0) = \Phi(\frac{\delta}{2} + \frac{\log\lambda}{\delta})$. In a similar fashion, we show that $P(d_0|H_1) = \Phi(\frac{\delta}{2} - \frac{\log\lambda}{\delta})$.

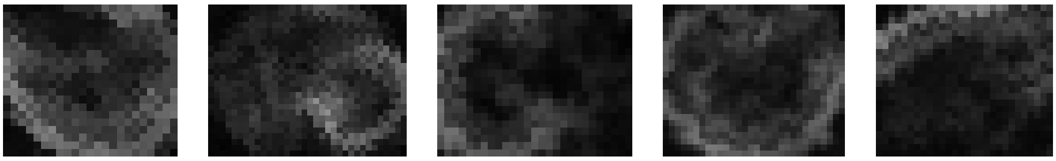
An invariance requirement would be appropriate for this problem. The knowledge of $\Sigma_i$ in the distribution of $I$ can be supplied by computing the variance of the background in a zone of the image without cells.

Stating $\Phi(\frac{\lambda'}{\|S\|_\Sigma}) = 0.05$, i.e. a confidence interval of 95%, gives $\lambda' = 1,64\|S\|_\Sigma$, hence

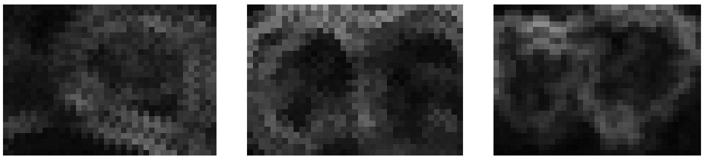
$$\sum_{i}^{n} S_{\cdot i}^T \Sigma_{\cdot i}^{-1} I_{\cdot i} \underset{d_1}{\overset{d_0}{\lessgtr}} 1,64\|S\|_\Sigma. \tag{2.35}$$

The aforementioned probability distributions are constructed by computing $\|S\|_\Sigma$ for image patches around each cell centre, and the threshold $\lambda'$ that separates the two distribution is decided as in Eq. 2.35 for the image. All centres that exceed this threshold are valid according to our maximum signal-to-noise criterion, and are then passed on to the next stage of the process as described in the following chapters. At this confidence, for the entire image set we validate an average of 82.95% of the cells per image. The remaining cell centres are therefore not taken into account for subsequent processing.

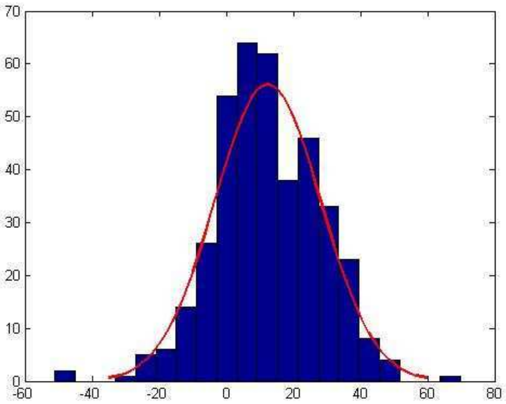
The application of the statistical test (eq. 2.35) to identify the cells in the image is illustrated in Fig. 2.20. In this figure, *imagettes* containing cells selected from a standard deviation image are shown against their $\ell$ scores. If $\ell > 0$, $H_1$ is in force and the presence of a cell is not rejected; on the contrary, if $\ell < 0$, this hypothesis is rejected. Figures 2.20.(a-d) examined the values of the $\ell$ criterion. Notice that the values of $\ell$ can be negative even if a cell is present for various reasons: when more than one cell is present (Fig. 2.20.g.h ), when the centre of the cell is not aligned with the centre of the template (Fig. 2.20.f) or simply due to the statistical uncertainty inherent to the test.



(a) $\ell = 1.06$ (b) $\ell = 5.92$ (c) $\ell = 3.83$ (d) $\ell = 21.58$ (e) $\ell = -1.38$



(f) $\ell = -1.45$ (g) $\ell = -15.16$ (h) $\ell = -5.60$



(i)

**Figure 2.20:** *Final scoring using (Eq. 2.35) (a-d)* $\ell > 0$, *imagette passes the test (e-h)* $\ell < 0$, *imagette is revoked. i) the distribution of* $\ell$ *showing its proportions with respect to* 0

## 2.5 Cell detection results and analysis

In this section we present a numerical assessment of the cell detection methodology along with a comparison with the current manual counting method as well as a method from the state of the art. For the benefit of better readability of the following analysis we first revisit counting performed manually, then present and discuss our counting results over the entire image data, and then present an analysis of errors that automatic counting is susceptible to produce, and finally finish by suggesting how validation of the counting helps in this regard.

### 2.5.1 Exploring manual counts

It is opportune to recall the counting method employed by our biologists because such a count not only establishes the standard an automatic method would aspire to achieve, but also recalls the $man - hour$ factor required to achieve that standard. Figure 2.21 shows the result for image 0032 from the manual detection utility we developed for labeling and therefore counting each type of cell, displayed in a different colour, present in an image.

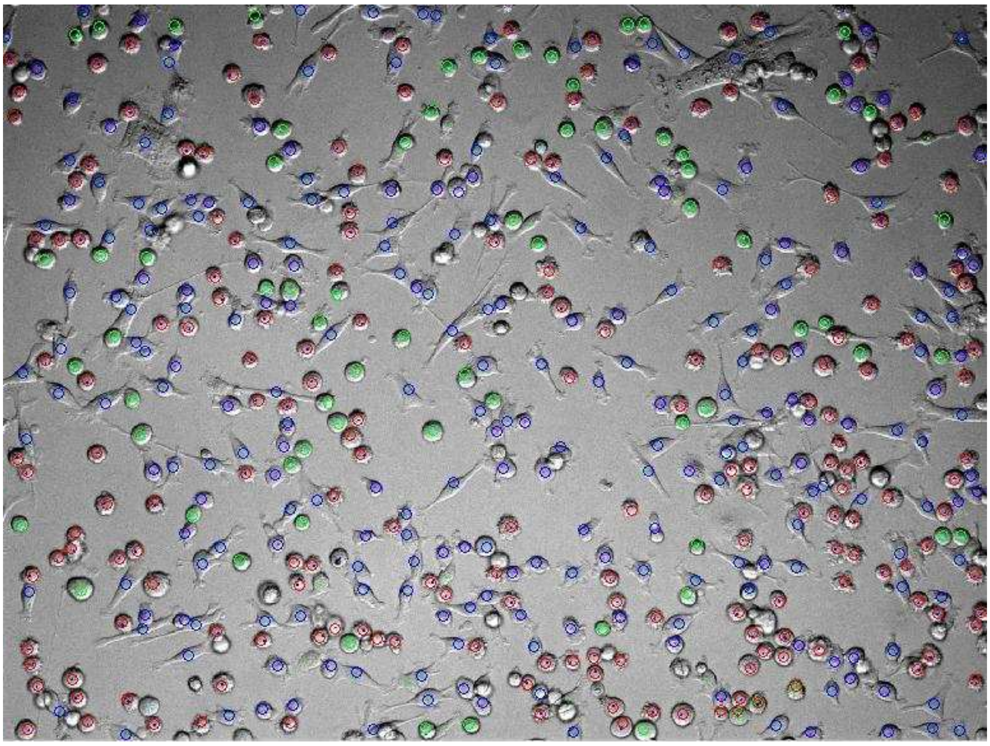


**Figure 2.21:** *Snapshot of image* 0032 *from the manual detection utility.*

Table 2.1 exposes more detail for this manual counting using our 14-image subset. Are presented here counting/labeling results for our two experts who we denote by

$E1$ and $E2$. It tells us that for the image set, about 12% cells are smooth round, around 30% blebbing and around 57% spread. The plenitude of spread and therefore harder-to-detect cells is a critical test for any automatic counting algorithm. The table is also a good reminder that manual counts as well as label attribution varies between experts. For instance, $E1$ counts 124 more cells than $E2$ for the image set, i.e. about 9 cells more per image. By the same token, each class attribution too varies by a few percents. The most marked difference of opinion between the experts is witnessed in image 113 which has a large number of cells in the course of splaying, where $E1$ counts 129 more spread cells than $E2$ and an equal number fewer in the two remaining classes. Therefore the 2% tolerance manifested between the two expert counts sets the counting algorithm a benchmark.

**Table 2.1:** *Comparison of cell counts obtained by two experts for each image in our 14-image sample dataset and for each class of cell.*

| Image | All | | Smooth round | | Blebbing | | spread | |
|---|---|---|---|---|---|---|---|---|
| | E1 | E2 | E1 | E2 | E1 | E2 | E1 | E2 |
| 0096 | 427 | 424 | 17 | 19 | 75 | 75 | 335 | 330 |
| 0100 | 302 | 301 | 17 | 17 | 45 | 45 | 240 | 239 |
| 0108 | 342 | 333 | 12 | 13 | 73 | 74 | 257 | 246 |
| 0112 | 456 | 445 | 37 | 37 | 108 | 107 | 311 | 301 |
| 0113 | 454 | 433 | 56 | 15 | 174 | 65 | 224 | 353 |
| 0032 | 528 | 509 | 75 | 72 | 192 | 190 | 261 | 247 |
| 0020 | 469 | 460 | 75 | 75 | 221 | 218 | 173 | 167 |
| 0021 | 710 | 703 | 164 | 163 | 269 | 268 | 277 | 272 |
| 0028 | 662 | 649 | 118 | 116 | 230 | 230 | 314 | 303 |
| 0073 | 368 | 363 | 56 | 55 | 120 | 129 | 192 | 179 |
| 0074 | 356 | 353 | 33 | 33 | 149 | 151 | 174 | 169 |
| 0082 | 624 | 612 | 100 | 100 | 151 | 153 | 373 | 359 |
| 0084 | 441 | 438 | 51 | 50 | 125 | 124 | 265 | 264 |
| Total | 6588 | 6464 | 875 | 826 | 2118 | 2012 | 3595 | 3626 |
| Mean | 470.57 | 461.71 | 62.50 | 59.00 | 151.29 | 143.71 | 256.79 | 259.00 |
| Std | 121.89 | 119.49 | 42.55 | 43.72 | 65.09 | 67.77 | 61.45 | 65.22 |

### 2.5.2 Automatic counts, and benchmarking them

We can now offer a breakdown of automatic counts obtained over the two subsets of the data images presented in Chapter 1 from the two experiments, namely “time effect” and “dosage effect”. We also present a comparison of the counting results with a state-of-the-art method. For this purpose we implemented Pinzón’s approach [162] alluded to in Section 2.3 that employes the Hough transform [89] as an easy-to-implement- method that could serve as a benchmark for the Halo transform. In this method, in order to locate a circular object, we draw at each point of the contour, straight lines perpendicular to the contour through this point. For a circle, all these lines intersect at the center of the circle. We use an accumulator parameter space, a matrix of the same size as the original image where all pixels are initialized to 0. We accumulate to perpendiculars over sampled points of the contours. It thus calculates, for each contour point, the equation of the line orthogonal to the contour, and increment on the accumulator, the value of all pixels on this line. Fig. 2.22 shows the result of such accumulation over several cell contours. In the end, the centers, intersections of many orthogonal lines, are traversed and binned more than other points. Local maxima detection, as before, is used to identify the positions of the centers, and therefore the presence of a cell.

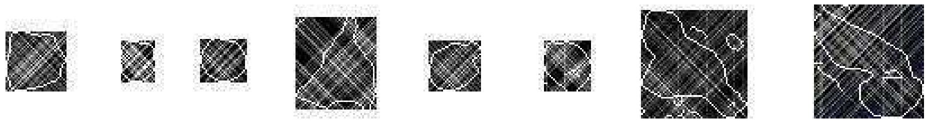


**Figure 2.22:** *Example of normals drawn (sparsely for sake of clarity) to cell walls adding to accumulator bins.*

A three-way comparison of the number of cells detected by the implementation of the Hough transform and by Halo filtering versus manual counting performed by expert $E1$ for our population of 142 images can be seen in Table 2.2. The manual counting establishes the yardstick to measure the efficiency of the two automatic methods. It is nonetheless important to know that it includes cells touching image borders, which are discounted later since it is difficult to conclude about their type. All automatic methods under-perform for the temporal effect series of cells since cells tend to elongate as environmental $PAI-1$ depletes. The manual counts presented are from expert $E1$, and *henceforth throughout the thesis the same counts and labels are used.* The choice of expert was made automatically since this is the expert we received counts initially and began the work with. The third data row presents the result of a single iteration of the filter. Subsequently, discs with the modal cell radius for the image, representing cells located around the detected centres are depletion from the image, and a second iteration of maxima detection is performed. This minute but portentous improvement in the application of the proposed filter results in an increase from about 86% to about 97% in cell detection, which suggests the significance of this iteration. Further iterations were empirically found to not add more than two new centres to an image, and in fact none at all in most cases, and were not pursued. As also can be discerned from the table, Hough transform manages about 94% of the cells while Halo filtering detects around 97%.

We conclude therefore that the method we propose performs better for the data and the conditions, and given its 3% shortcoming is extremely close to the 2% deviation already existing between the two experts, can be used to relieve them of the job.

**Table 2.2:** *Comparison of cell detection performances: ground truth (human expert) vs Hough Transform vs Halo Transform.*

| *Number of cells detected* | Cellular series: Temporal effect | Cellular series: Dosage effect | Total |
|---|---|---|---|
| Human expert | 33651 | 33250 | 66901 |
| Hough transform | 30252<br>**89,90%** | 32646<br>**98,18%** | 62898<br>**94.02%** |
| Halo transform<br>*iter. 1* | 26947<br>**80.08%** | 30531<br>**91.82%** | 62898<br>**85.92%** |
| Halo transform,<br>*iter. 2* | 31902<br>**94.8%** | 32975<br>**99.17%** | 64877<br>**96.98%** |

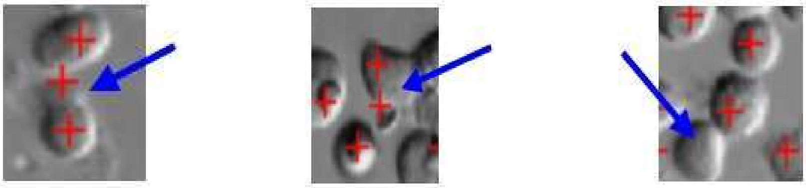
### 2.5.3 Error analysis

The figures presented in the previous sub-section represent brute counts, irrespective of the errors committed in the counting. In Table 2.3 we look at the counting errors over a sample of 14 varied images from our set of 142; for which the human expert has counted 6588 cells. We find that errors may occur

- in the localization of a local maximum - it might appear in the background or the border of the image and not on the cell itself (Fig. 2.23 *a.*);
- in the number of local maxima - multiple ones might be found on the same cell (Fig. 2.23 *b*), either due to interaction with neighbouring cells in an agglomerate or due to its disproportionate size with respect to other cells;
- or the absence of a local maximum (Fig. 2.23 *c.*), again due to interactions with other cells in a cluster.

**Table 2.3:** *Error analysis of the Halo transform counting for* 6588 *cells.*

| Cells detected | Undetected round cells | Undetected spread cells | Multiple detection | Detection in background |
|---|---|---|---|---|
| Number | 327 | 394 | 91 | 257 |
| % | 4.96 | 5.98 | 1.38 | 3.90 |
| Mean | 23.36 | 28.14 | 6.5 | 18.36 |



(a) (b) (c)

**Figure 2.23:** *The three types of error committed by the cell detection algorithm.*

The comparison of the filtering algorithm presented with a simple filtering shows that they have been necessary to limit false positives i.e. missed cells. A false positive could be a dangerous prospect since it could ultimately lead to the diagnosis of absence of aggressive blebbing metastase when it was present. The algorithm has shown to miss only about 5% of all round cells i.e about 3.5% blebbing cells, this repsenting the risk factor of a blebbing false positive. The first type of error represents false negatives, that is to say, cells are counted where they are absent. These are particularly inconvenient for automatic algorithms that we use in the following work because a false centre detected between two cells cannot help create a segmented object representing a cell itself, and rather it interferes in correct centre-guided segmentation of the neighbouring cells. This produces segmented objects with which cannot be characterized by a classification algorithm into any of the three classes of cells. The hypothesis testing mechanism we discuss in section 2.4 has been put into place in order to reduce the risk of false negatives, and help the processing chain to proceed smoothly. Through it we surmise for any given image that the test allows us 90% confidence that an average of 82.95% will not be false negatives. These validated centres can therefore be used to produce desired cellular characteristics leading to correct identification of cells into the pertinent types.

## 2.6 Conclusions

This chapter was dedicated to the first tangible and in its right a crucial step within the framework of our problem. A new method for automatic counting of “in vitro” cells, well adapted to microscopy of cellular suspensions, is presented. Counting results show that the proposed filter detects about 97% of the cells and commits few errors, ensuring that cells identified are mostly in agreement with reality. These results were obtained from nearly 150 images in a difficult context, and acquired in a non-standard environment. We propose adaptive pre-processing steps for the rectification of these defects. This work essentially helps experts identify cells in less time and effort; the automatic count is only about 3% smaller than manual counts obtained by experts, given also that manual counts vary among themselves by 2%.

The work accomplished as described in this chapter resolves part of the most tedious routine our partner biological experts had to engage in, that is, count all cells in a microscopy image by visual inspection. By visual inspection of such images, it is possible to make a qualitative evaluation of the signals representing cells, but even such a simple thing as saying if a signal is present or not becomes strongly biased by the user, and the measure is difficult to reproduce. Fully automated methods for quantitative evaluation of digital image data are therefore vital for its robust and reliable evaluation. The described signal-validation process increases this robustness. There are limitations to this method in that it supposes a rigid signal template and non-correlation between its column data. A further improvement that uses no prior knowledge has been envisaged for future work and offers possibility of a higher percentage of cell-validation.

However, manual counting till this stage in the thesis offers the benefit of on-line cell classification, i.e., the experts are also able to decide while counting which cell should be labeled into which category, with a fair amount of overlapping decisions. We however, will have to inspect the aspect of each cell obtained in this step in order to arrive at that decision. This necessitates the need to cut each cell out of the image, in other words, to perform image segmentation. Only when this segmentation is accurate enough that we would be able to calculate classification attributes that could help us arrive at a decision. These two steps are described in the following chapters.

# 3

# Pursuing a relevant segmentation

# Preamble

For any image segmentation problem, the knowledge of the "segmentation goal" is very crucial [40] because in a context where the number of regions to extract is not known *a priori*, there is generally no unique solution to the division of an image into "relevant" tessellations. The segmentation goal in the application scenario was to separate cells from the background and among themselves. The first was achieved in the previous chapter. This chapter revolves around the objective of separating cells from other cells in a situation where cells tend to adhere, and to hone the initially limited accuracy of the separation of cells from the background. Let us begin by recapitulating where we left off at the end of last two chapters. Thus far we have located the cells on an image and found a way to cut out parts of the image that principally comprise pixels belonging to cells, which we call *connected components*. This chapter takes us through the segmentation mechanism we have devised to separate cells among them. Having done this, we would be able to calculate various classification attributes from the individual cell images. Thus the data we exploit from the previous steps of the processing chain comprises: the original grey level image, the binarised image of agglomerate connected components, and the location of the centres detected and validated for the cells in the image.

## 3.1 Image segmentation

*Image segmentation*, or simply *segmentation*, is one of the most ubiquitous and challenging problem encountered in image processing, referring to the task of detecting boundaries of objects of interest in an image, or alternatively to the image processing operation which gathers pixels among themselves according to predetermined criteria [153]. Each group of pixels then forms a *region* or a *segment*, and a paving of the image by regions is thus obtained. Segmentation is often the most vital and most difficult step in an image analysis task. The segmentation result usually determines eventual success of the analysis. For this reason, many segmentation techniques have been developed, and there exist almost as many segmentation algorithms [241] as there are segmentation problems.

To understand image segmentation we need to realize that it is an application of the *set partitioning problem.* A partition of a set $X$ is a division of $X$ into non-overlapping and non-empty *parts* or *blocks* that cover all of $X$. More formally, these *partitions* are both *collectively exhaustive* and *mutually exclusive* with respect to the set being partitioned.

A partition of a set $X$ is a set of nonempty subsets of $X$ such that every element $x$ in $X$ is in exactly one of these subsets.

Equivalently, a set $P$ of nonempty sets is a partition of $X$ if

1. the union of the elements of $P$ is equal to $X$. (the elements of $P$ are said to *cover* $X$): $\bigcup P = X$
2. the intersection of any two distinct elements of $P$ is empty. (all elements of $P$ are pairwise disjoint): $A \cap B = \varnothing$ if $A \in P$, $B \in P$, $A \neq B$

where $\varnothing$ is the empty set.

The notion of the similarity or the dissimilarity criterion augments set partitioning to apply it to image data because the *a priori* definition of what constitutes the elements of the set is not implicit. The choice of the criterion translates homogeneous regions into partitions or segments. Henceforth, defining image segmentation is straightforward: The segmentation $D = \Lambda D_i(I)$ decomposes set $E$ into zones that are:

- disjoint and cover the whole space $E$,
- where the image function $I$ is homogeneous according to criterion $\epsilon$.

## 3.2 Segmentation of cellular images

Segmentation algorithms and techniques are remarkable in their numerousness and variation an the literature abounds with algorithms. However, there does not appear to be any unifying principle guiding many of them. Some are one dimensional signal processing techniques which have been extended to two dimensions such as pattern recognition. Many are the same basic algorithm with parameter values tweaked to suit the problem at hand. Alternatively, the parameters are optimized with respect to a suitable training set, without thought on how to vary them for images with different properties. Moreover producing an exhaustive list will exceed the scope of this document. Nonetheless, there are some general themes grouping methods of segmentation, in a dichotomous one-versus-the-other manner, sometimes exploiting the same mathematical bases. These themes provide a nice categorization of segmentation methods along the lines of paradigms of approach to a problem. The following reviews these.

The difficulties in automatic segmentation of images of cell nuclei in tissue produced by fluorescence microscopy usually have three causes. First, the image background intensity is often uneven due to auto-fluorescence from the tissue and fluorescence from out-of-focus objects. This unevenness makes the separation of objects and background a non-trivial task. Second, intensity variations within the nuclei further complicate the segmentation as each nucleus may be split into more than one object, leading to over-segmentation. Third, cell nuclei are often clustered, making it difficult to separate the individual nuclei.

Despite active research in the field, cell segmentation remains a challenging problem due to the diversity and complexity of microscopy images across cell types and application contexts, with weak contrast, touching nuclei, diffused background and varying size and shape of cell nuclei all posing challenges to existing methods. Generality and portability of existing methods is also a challenge.

One of the first and largest application fields for image analysis is biomedicine in general and, particularly, pathology. The final diagnostic decision for many diseases is based on microscopic examination of cells and tissues. In particular, cancer is always finally diagnosed through the microscope. Therefore, ever since the first appearance of computers, significant development efforts have been aiming at supplementing or replacing human visual inspection with computer analysis [12].

In addition to its diagnosis and screening applications, microscopic examination can help in grading the cancer, i.e., to determine how aggressive it is and how it should be treated. The grading may be based on an assessment of the amount of DNA per cell nucleus or on a specific staining of some other marker/pigment, e.g., associated with cell proliferation. Computerized image analysis may be used to automatically measure the staining reaction in a quantitative way i.e. by the number of stained objects and their area [167].

In order to study and control stem cell multipotency and self-renewal, cell migration and cell division have to be modeled. This is done by tracking individual cells over time. Image sequences are acquired using time-lapse microscopy supplemented by an incubator system such as ours (refer to Chapter 1. 1.5.3.B) to keep the cells alive and to make it possible to investigate and compare multiple cell cultures, e.g., manipulated cells and controls. Tracking the cells in these sequences presents a number of image analysis challenges. Cells may touch, overlap, or enter or leave the microscopic field of view, as well as divide or die during the time-lapse experiment. The robustness and accuracy of automated tracking methods can be improved if statistical *a priori* information on typical cell movement patterns can be incorporated into the tracking algorithm [77].

Cells are elaborately subdivided into functionally distinct, membrane-bounded compartments. Each compartment contains its own characteristic set of specialized molecules, and complex distribution systems transport specific products from one compartment to another or from one compartment and out of the cell. Specific proteins within living cells can be tagged with green fluorescent protein and similar tags, by genetic engineering. The localization and movement of the tagged protein can thereafter be imaged as “blobs” using fluorescence microscopy. Methods for the detection, characterization, quantification, and positioning of blobs are required. In addition these kinds of methods are also of great interest to the pharmaceutical industry in the development of new drugs [121].

The difficulty of the segmentation problem highly depends on the type of specimen to be analyzed. If we are dealing with cytological specimens where the cells are singly lying on a clean background with well-stained nuclei, and if the analysis task is limited to nuclear properties, then a simple automatic thresholding method may be sufficient. If, on the other hand, the cells are presented in intact tissue, such as histopathological tissue sections, and the nuclei as well as the cytoplasms are stained, then the segmentation task may be difficult indeed. Sometimes, it is necessary to use interactive techniques in order to obtain sufficient data quality. But even in these cases, it is useful to push the automated segmentation as far as possible.

So far, no general standard solution to the segmentation problem has been found. A new tailored solution is typically developed for each application problem. Still, these solutions can be discussed in terms of image and object models, i.e., what kind of information in the images they are based on and whether it is mainly intensity, edges, connectivity, or shape. Through such a discussion, as we attempt in this section, some general properties can be seen and, hopefully, some useful steps towards more generally useful segmentation approaches can be taken.

In a general, application-independent manner, Robert M. Haralick and Linda G. Shapiro classified the image segmentation techniques as: *measurement space guided spatial clustering*, *single linkage region growing schemes*, *hybrid linkage region growing schemes*, *centroid linkage region growing schemes*, *spatial clustering schemes*, and *split and merge schemes* [83]. Most of these algorithms were conceived for greyscale images, and some have been extended to colour images. In the domain of cell images, the same classification persists and can be seen in various image processing suites of methods developed for both specific and relatively general problems.

In the following we will discuss some of the broad categories of methods.

### 3.2.1 Thresholding and pixel-classification

A simple and often used method for image segmentation is thresholding based on histogram characteristics of the pixel intensities of the image. Here, it is implied that objects of interest are brighter or darker than other parts of the image. For an overview of thresholding techniques, see [174]. In order to get a satisfactory segmentation by thresholding, a sufficiently uniform background is required. Many background correction techniques exist (see, e.g., [105, 182]), but they may not always result in an image suitable for further analysis by thresholding. The transition between object and background may be diffuse, making an optimal threshold level difficult to find. At the same time, a small change in the threshold level may have a great impact on further analysis. Feature measures, such as area, volume, mean pixel intensity, etc., directly depend on the threshold. These effects can be reduced through the use of fuzzy techniques, e.g., fuzzy feature measures [147]. Adaptive thresholding, i.e., local automatic thresholding, can be used to circumvent the problem of varying background or to refine a coarse global threshold [46]. The problems of segmenting clustered objects and choosing a suitable threshold level for objects with unsharp edges will, however, remain. Thresholding does not have to be the final step in the segmentation procedure. Usually, thresholding on its own is not powerful enough a tool to segment cell images, except when the cells are completely separate of each other. It is, however, the first pass that simplifies the second pass in a two-pass segmentation. An intensity threshold can be used as a start for further processing (as Fig. 1.8 illustrated), e.g., by the morphological operations presented below and/or visual inspection. For example, thresholding is combined with region growing in [199] to obtain a binarisation and then restrict the regions to blood cells using regional context information. Still, some authors do find thresholding sufficient to extract cell matter from the image if the separation among cells is not required, or as in [127] and [8] where only cell nuclei are of interest.

Thresholding is essentially a classification method for the grouping of pixels sharing certain similarities to select the pixels belonging to the objects. Methods for selecting the threshold once the similarity criterion is defined can be as varied as methods for data classification, either supervised or unsupervised. Simple methods of threshold selection that classifies pixels into two classes could be traced back to Isodata [7] that iteratively refines the threshold akin to $k$-means and Otsu [152] that tries to maximize intra-class variance, and can be used to separate more than two classes [117]. This classification may be done by measuring a set of features at each point and defining a decision surface in the feature space [154]. For colour images, multi-class pixel-classification becomes interesting because sub-cellular structures may be pigmented variously. Lebrun et al. [111] use an $SVM$ to learn various sub-cellular structures using among others colour information to achieve an accurate separation of nuclei, cytoplasm and image background. Lezoray et al. flirt with the idea of combining pixel-classification with watershed segmentation beginning with [115] by first aggregating watershed markers based on colour using $k$-means and Bayesian classification. This however necessitates defining a colour gradient measure as the topographic function for the watershed. Meurie et al. [138] combine pixel-classification with definition of one class as watershed markers. Multi-channel cellular images necessitate additional requirements of histogram-concordance [114] since the initial pixel classification may assign same pixels different labels in different colour channels.

### 3.2.2 Edge-Based Segmentation

Another observation used as a basis for segmentation is that cells are surrounded by edges where the intensity changes rapidly. Anoraganingrum [41] performed edge detection on melanoma cells using median filtering and mathematical morphology. Edges are usually initially extracted as a gradient image in which the local maxima are connected. A drawback of this method is that one often runs into problems when trying to produce closed curves. A powerful solution to this problem is offered by the so-called snakes or active shape models [97]. From a rough marking of the border or a seed inside the object of interest, a curve expands until it finds a strong edge. The function describing the expansion consists of different energy terms attracting the curve to edges [190]. Problems with this model consist in defining suitable energy terms and, again, constructing automatic seeding methods that are restricted to one unique seed per nucleus. Edge-based snakes, although very successful in the bio-medical image segmentation in general such as $MRI$ images [37] used in order to precisely segment one object, have not caught on as means to segment cell images with a large number of cells, absence of colour and small grey level distinction as in our images because they tend to stray and fit themselves to neighbouring cells or are simply computationally too expensive for a large image with many objects to locate. Few authors such as [38] do use them when the image contains about a dozen cells and the object is actually the fairly distinctive cell nucleus. Cell nuclei are usually convex and fairly round or elliptic. The shape of the cell nuclei itself can therefore be used as part of the object model. In [229], a $3D$ blob segmentation method based on elliptic feature calculation, convex hull computations, and size discrimination is described. Careful choice of a scale parameter is needed, and the edges of the resulting objects will not necessarily be aligned with the edges of the nuclei. A restricted convex hull is computed for slice-wise $3D$ segmentation in [11]. Distance information in the restricted convex deficiency defines the location of separating lines between clustered objects. The information obtained per slice is later joined to construct $3D$ objects. Watershed segmentation applied to distance transformed binary images (usually binarised through thresholding) is useful for separating touching objects that are convex (see [130, 46, 220]). In [105], similar separating lines between touching objects are found in a slightly different way. The distance image is thresholded, creating a new binary image consisting of shrunk versions of all the objects. Dividing lines are thereafter defined as the skeleton. Including the time dimension, Zimmer et al. [243] suggest the segmentation and racking of motile cells using a parametric active contour model, along with a comprehensive strategy of working with cellular images. io et al.[187] addressed the problem of parasitemia estimation using edge detection and splitting of large clumps made up from erythrocytes. The outcome of the approach was shown to be satisfactory for well-stained samples with well-separated erythrocytes. For the same problem, watershed transform [216] were also employed, given that local maxima indicate the centers of convex shapes, i.e. blood components particularly erythrocytes. This concept, however, is only justifiable for images which exhibit a small degree of cell overlap.

### 3.2.3 Region growing and other region-based methods

If we model the objects as consisting of connected regions of similar pixels, we obtain region growing methods. The name comes from the fact that starting regions grow by connecting neighboring pixels/voxels of similar gray level.

Many region-growing algorithms result in over-segmented images; i.e., too many object regions are formed. In [157], region growing is combined with region merging based on edge information, and in [120], the images are preprocessed with an adaptive anisotropic filter to reduce over-segmentation. The adaptive anisotropic filter reduces noise in homogeneous regions while sharpening discontinuities. Using these methods, one is still left to face the prominent clustering problem, i.e., finding separation lines when no intensity variation is present. Another approach, described in [2], is to let the regions grow from predefined small regions, known as seeds. Each region in the resulting segmented image will contain exactly one of the starting seeds.

Both manually marked seeds and an automatic seeding method are described. The problem with this approach to cell nuclei segmentation is that it is very difficult to construct a seeding method that puts exactly one seed in each nucleus, even when the nuclei are clustered and/or have internal intensity variations. Garrido et al. [66] approximated red blood cell locations using a parametric ellipse model to generally fit the cell and refined its contours using a region-based deformable model as introduced by Chan and Vese [34] that exploits region homogeneities. This region-based snake model is often successfully used in the segmentation of images of non-adherent blood cells. For example, region growing is combined with thresholding in [199] to obtain a binarisation and then restrict the regions to blood cells using regional context information in the deformable model. Eom et al. [55] use region-based active contour model, where region information is estimated using a statistical analysis, to segment blood cells.

#### 3.2.3.A Watershed Segmentation

A popular region-growing method, which has proved very useful in many areas of image segmentation and analysis, is the so-called watershed algorithm. The *watershed* algorithm [109, 19] is a morphology-based segmentation method [140]. It is based on the assumption that any greyscale image can be considered as a topographic surface [20]. If we flood this surface from its minima preventing the merge of the waters coming from different sources, the surface is eventually separated as two different sets: the catchment basins and the watershed lines. If we apply this transformation to the magnitude of image gradient $||\nabla I||$ , the catchment basins correspond to the uniform sub-regions in the image and the watershed lines correspond to the edges. We will skip a detailed discussion of the watershed transform in this section because we shall be seeing a lot more of it later in the chapter.

The concept of watersheds comes from the field of topography, referring to the division of a landscape in several basins or water catchment areas. A good example is the continental divide that separates the USA into two main regions: one associated with the Atlantic Ocean, and another associated with the Pacific Ocean. So, on rainy days, all the drops of rain that fall on one side of the divide flow into one ocean, while rain falling on the other side of the division will flow into the other ocean. It is clear that the water will reach the ocean provided that it is not trapped in a *local minimum* along the way.

Beucher and Lantuejoul used it to segment images of bubbles and $SEM$ metallographic pictures. Unfortunately, this transformation very often leads to an over-segmentation of the image. To overcome this problem, a strategy has been proposed by Meyer and Beucher [140]. This strategy is called *marker-controlled watershed segmentation*. This approach is based on the idea that machine vision systems often roughly “know” from other sources the location of the objects to be segmented.

A gradient image is often used as the topographical function for the watershed transformation, because the main criterion of the segmentation is the homogeneity of the greylevels of the objects present in the image. But, when other criteria are relevant, other functions can be used. In particular, when the segmentation is based on the shape of the objects, the distance function is very helpful. Section 3.5 discusses, among some other things, our efforts at exploring several topographic functions in order to arrive at the suitable one.

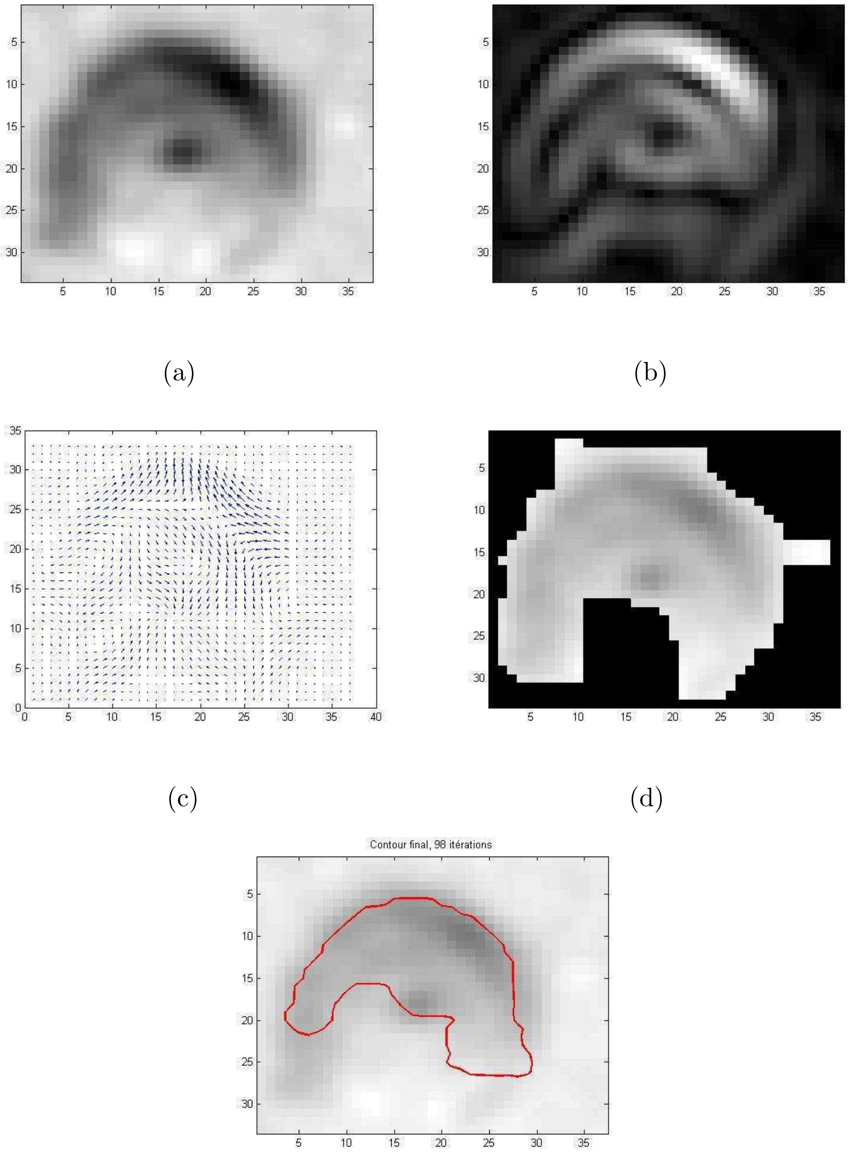
**Concluding remarks**

Often none of the above-described methods will alone produce a satisfactory result on the more difficult types of cell and tissue images. We may, for instance, have problems if 1) the cells are clustered, 2) the image background varies, and 3) there are intensity variations within the cells. By combining the methods, more powerful models of the situation can be created which can solve many segmentation problems. The amount of literature favouring the watershed, and especially the propensity of authors to use the watershed transform in problems involving cell images, has shown that the watershed approach is a useful core component in such more complex segmentation models. Hence it becomes a default choice as segmentation method for another of these segmentation problems.

### 3.2.4 Watershed Segmentation as our method of choice

As we showed in Chapter 2, the images suffer from highly directional illumination, which causes image gradients to be lopsided and inaccurate as means to represent image variability, thus obviating the use of a gradient-based segmentation technique. With these as well as the adherent nature of the cells and their textural similarity in mind, simple thresholding techniques that exploit pixel intensities nor more advanced gradient-based edge-detection methods like active contours can be deemed appropriate for separating cells among one another. In fact none of the methods that use the gradient or other local image variability measures such as image statistics or texture measures resulted in well-formed cells, for the reason that though the global illumination gradient has been mitigated, the local effects of directional illumination can not be and must not be eliminated from the images. It is these that make the cell silhouettes and the textured cell interior visible by allowing parts of the image to be lit up and others be darkened by shadowed, but it is also these that impose on us a view by anamorphosis of the data, that is, the light and the dark parts of the image do not accurately coincide with cellular structures, e.g. the cell wall and in fact all the edge pixels are mal-positioned, because the cell wall is lighted up on one side causing lighter pixels to extend into the cell and on the opposite side shadows extend well beyond the cell. This information is difficult to reconcile and the detected cell contour lies inside the cell on the brighter side and outside the cell on the darker side. Any mechanism that establishes the position of the cell wall by using local information therefore positions the edge at the wrong place. Similarly, in histogram-thresholding the two cell sides go to the two opposite extremes of the image histogram and therefore to different classes. More sophisticated pixel classification, such as that using either one local information e.g. the local standard deviations nor those using a combination of local information such as in [154] resolve the problem of incorrect assignment of pixels in classes. One such approach using partial-membership assignments was successful in separating the background, the most high-local-variability parts of cells and the remainder of the cell pixels, but these assignments were still affected by the problems associated with locally directional illumination. This is incongruous with the segmentation goals, and a combination of the classes is equivalent to a reversion to the two-class Otsu thresholding 2.2.2 explained in the previous chapter. Still, it was found to be aid in the application of the segmentation method actually selected and is therefore mentioned in Section 3.5.4. Fig. 3.1 elucidates these problems using segmentation methods implemented from the literature.

Obviously a region-based approach is required. Of these, those that try to group pixels into individual cells using pixel-similarity measures, such as region growing, fail for the same reasons as above. Region-based snakes are geared towards the search of image parts that exhibit homogeneity, tantamount to pixel-similarity, and are computationally exorbitant when used on large images such as ours. The watershed transform offers an interesting alternative because it can be independent of local image information if such a formulation is feasible. It also guarantees closed object contours and a degree of salvageability of objects of interest, that is to say for example that the lighter corner of a cell may form a separate region but algorithms as we shall see in Chapter 4 exist that could regroup them. Among all the segmentation methods, the watershed transform is certainly one of the most popular judging by the great diversity of applications in which the method appears relevant. Several factors make it such a success. The morphological segmentation paradigm is a two steps procedure: in the first step, the image is analyzed and some germs



(a)

(b)

(c)

(d)

(e)

**Figure 3.1:** *Zoom on one cell of the image and a review of the implementation of some classical segmentation methods: (a) the cell tile in question (b) the norm of the two partial derivatives showing the gradient, (c) local dominant direction of the gradient vectors not pointing in a uniform direction, (d) a hysteresis thresholded connected component, and (e) a gradient-seeking active contour eating into the cell.*

or markers are introduced within each object of interest; the precise location of the marker inside the object has no importance. This determining step may also be done iteratively or interactively, permitting an elegant and easier decision procedure. The second stage of the segmentation consists in finding the strongest and best contours separating the markers and this part is independent of any parameter setting and fully automatic. Furthermore, a very efficient computation method exists for the watershed transform, making the whole strategy one of the least expensive ones in terms of computation time. Last, the strategy is versatile. The watershed transform is well adapted to many segmentation problems. It may be also used on more complex structures such as neighborhood graphs or trees. It also offers nice perspectives in multi-scale segmentation strategies and user-interactive segmentation schemes. The ubiquity of the watershed transform as a means for image segmentation, particularly in application to biological images, makes it an attractive prospect for a segmentation application such as ours, and offers a vast array of competing applications for reference and comparison. The watershed transform, in addition to being a region-based method, brings the added advantage of being not tied to a particular image representation such as grey level or indeed gradient, and stands out as the method of choice for our application.

In this chapter, the mechanisms of this transformation is briefly recalled and in the associated annexes some major contributions are listed: formal definitions, computation methods and connected works. The main focus remains on the examination of the watershed transform as an organic part of our application, without having to wrap the principal segmentation in a panoply of post-processing methods not related to morphological image processing.

Before presenting the segmentation methods we utilize in this chapter, let us take a brief detour to define the manner in which we evaluate the results those segmentation methods produce.

## 3.3 How good is a segmentation: Segmentation Quality Evaluation

Prior to presenting any number of competing segmentation schemes it is necessary to have a way to perform discriminatory assessment of the quality if each segmentation. In this section we present some existing ideas and a natural adaptation of one of them to our situation.

Evaluating a segmentation method is at the same time necessary, important, and ill-posed. It is necessary to rank segmentation results obtained by different methods, or one method with different sets of parameter values, in order to select a method or to tune parameters. Although this sounds natural, deciding on a way to rank results can actually be difficult. Evaluating a method is also important in order to assess and validate it: this requires to have a quantitative and qualitative measure of the segmentation results. Here again, deciding on a metric to perform these measures is not obvious. Finally, evaluation in itself is an ill-posed task. It refers to different criteria, some of which have various interpretations and are measured with various metrics.

### 3.3.1 Methods of segmentation quality evaluation

Because of the profusion of image segmentation methods evaluation becomes crucial, but the problem of defining a good segmentation remains unsolved and the solution mainly depends on the goal. A good segmentation can be defined as a segmentation true to one given by a human being.

The criteria of quantitative evaluation can be split into two classes, depending whether we possess or not a *ground-truth* which constitutes a reference segmentation. This reference is directly accessible in the case of computer generated, i.e. the segmentation which was used for synthesizing the image, but in the case of real images it must generally be built “by hand” [161] by an expert of the application domain. For an evaluation on real images the notion of “segmentation goal” is very important [40] because in a context where the number of regions to extract is not known *a priori*, there is generally no unique solution to the division of an image into “relevant” regions. The “relevance” of a region is indeed a notion highly dependent on what Correia and Pereira in the same article call an “application scenario”. Since for our purposes such a ground-reality is existent we shall limit ourselves to the first class of methods.

Similar to the segmentation theory itself, there is no established standard procedure for the evaluation of its results. In literature there exists a multitude of very different approaches. A general classification of evaluation methods has been proposed by Zhang [239], categorising three variants: *analytic methods*, *empirical goodness methods*, and *empirical discrepancy methods.* In recent studies, empirical goodness methods are also referred to as *unsupervised evaluation methods*, empirical discrepancy methods are denoted as *supervised or stand-alone evaluation methods* e.g. Zhang et al. [238].

Most existing approaches in practice are supervised methods using discrepancy measures between a reference and the segmentation, based on a qualitative visual and a quantitative evaluation based on geometrical features of the segments, e.g. area, perimeter, and shape, using manually derived ground truth. Many criteria of "discrepancy" have been proposed, which can be used when a ground truth is available, usually given by an expert of the application domain, who is supposed to exactly know what he is expecting, in terms of accuracy, level of detail, etc. Among these discrepancy criteria we could cite *Vinet measure* [217], the measure of Yasnoff et al. [230] which counts the number of mis-segmented pixels, the Baddeley distance [225], and the ultimate measurement accuracy of Zhang [240]. The measure of consistency between segmentations of Martin [136] can also be used as a discrepancy measure between a segmented image and a reference image.

Statistical model-based methods have also been explored, for example [244] examine the Dice Similarity Coefficient ($DSC$), a statistical validation method based on the spatial overlap between two sets of segmentations of the same anatomy. $DSC$ values are computed and logit-transformed values are compared in the mean with the analysis of variance ($ANOVA$).

### 3.3.2 The discrepancy criterion

Recent publications have clarified the different evaluation criteria. Zhang describes three types of evaluation, namely analysis, goodness, and discrepancy. *Analysis* focuses on the algorithm used in a segmentation method, in particular its complexity, in terms of memory or runtime. This type of evaluation is important to assess the performance of an algorithm, or to implement optimisation. In our context however, analysis is not the main evaluation required: runtime, memory allocation or optimisation are secondary for an experimental study. *Goodness* evaluates results based on image and object properties, regardless of external references. It can be based on intra-region uniformity, inter-region contrast, or region shapes. These somehow artificial criteria have been used to reproduce human judgment on the quality of segmentation in itself. They do not apply in our task however: our goal is not to produce visually pleasing results, but to accurately segment adhering cells. The final criterion by Zhang is *discrepancy*: this type of evaluation consists in comparing results against references. It requires prior referencing of the images, and a measure to compare a computed segmentation against a reference segmentation. It allows an objective comparison of segmentation methods and guidelines for tuning a method's parameters, and therefore this is the evaluation required in our context.

We use a supervised method where the ground-truth is obtained by delimiting cell boundaries by hand, and the resulting contour and the enclosed region are used as references, using which different segmentations methods can be compared. What does not change is the nature of the segmentation method, i.e. a watershed mosaic of regions. This has the implication that all pixels belonging to the image are accounted for, by falling either within one region or catchment basin or into the background. The ground-truth serves to establish whether the pixel is in the correct basin or is elsewhere. Thus a group of pixels that should have been, according to the ground-truth, in a particular basin, may either be attributed by a segmentation to another, or just as well the other way round. This gives rise to two types of errors in segmentation that the evaluation tries to quantify: *basin overflow* (*BO*) beyond cell boundaries, resulting in excessive pixels attributed to a cell; and *basin shortfall* (*BS*), i.e. loss of cell pixels to the background or neighbouring basins. Basin overflow is calculated as the ratio of segmented region pixels lying outside of the manual contour, and basin shortfall is calculated as the ratio of the manually cut region pixels not in the segmented region.

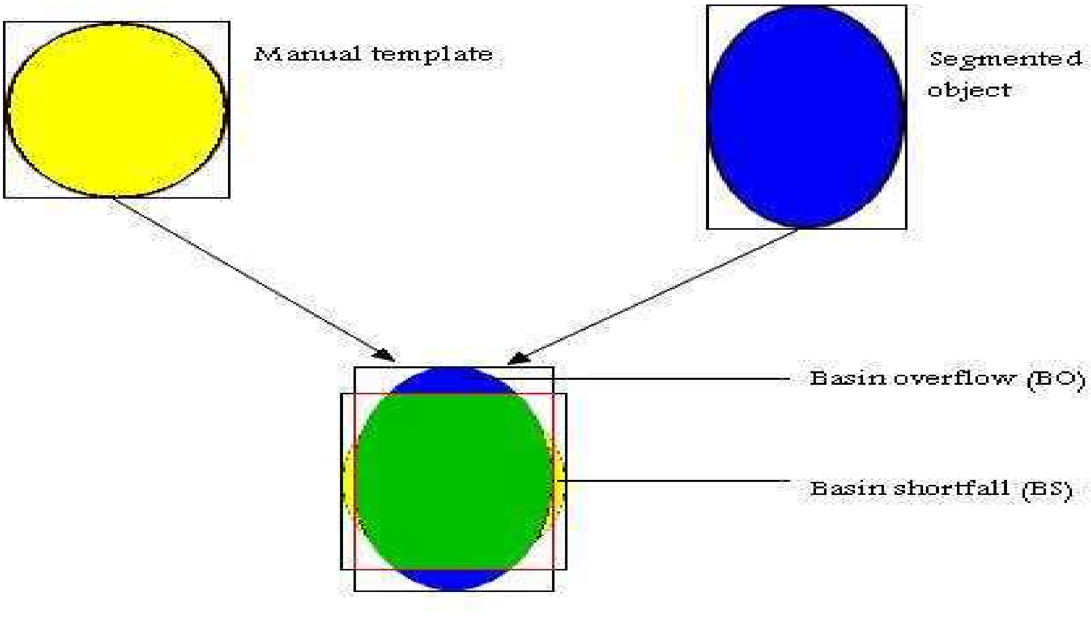


**Figure 3.2:** *Illustration of the comparison.*

In reality, this discrepancy criterion was chosen for the reason because it mimics the way the human mind discerns the quality of a segmentation (see Section 3.3.3). This is a simple yet handy criterion to evaluate the discrepancy to the ground-truth of any segmentation method in our application scenario, and we shall see in later sections, produces fairly discriminating results. In latter text, we sometimes refer to Segmentation Quality Evaluation by the abbreviation *SQE*.

### 3.3.3 The qualitative criterion

Segmentation quality is obviously describable in a qualitative manner than only a quantitative one. The human eye is better than numerical criteria for discerning the accuracy of a segmentation. Therefore we augment our discrepancy criterion with the anvil of human visual inspection. Through this visual scoring we conclude the quality of a segmentation based in much the same manner as we described earlier i.e. as $CA$, $BO$ and $BS$ or how much of a cell is well-segmented, how much goes missing and how much of it engulfs parts of other cells; as well as the precision with which segmentation cuts or connected component edges sit on object boundaries. The last factor is possibly decisive because it is absent in the discrepancy criterion.

The decision is arrived at by first studying the manual segmentation and then looking at each individual segmentation candidate for $SQE$. Poor segmentation can be easily spotted thanks to objects missing chunks or encroaching others or the background. Ranking close segmentations, particularly in order to determine the best among the candidates, necessitates closer inspection, in finer cases of the validity of the object contours. We generally score each segmentation on a scale of 1 to 10, with 10 being a perfect superposition with the manual segmentation. Even though not graphically representable, this criterion represents a factor that could nullify the numerical comparative edge of one method over another.

### 3.3.4 Evaluation Methodology

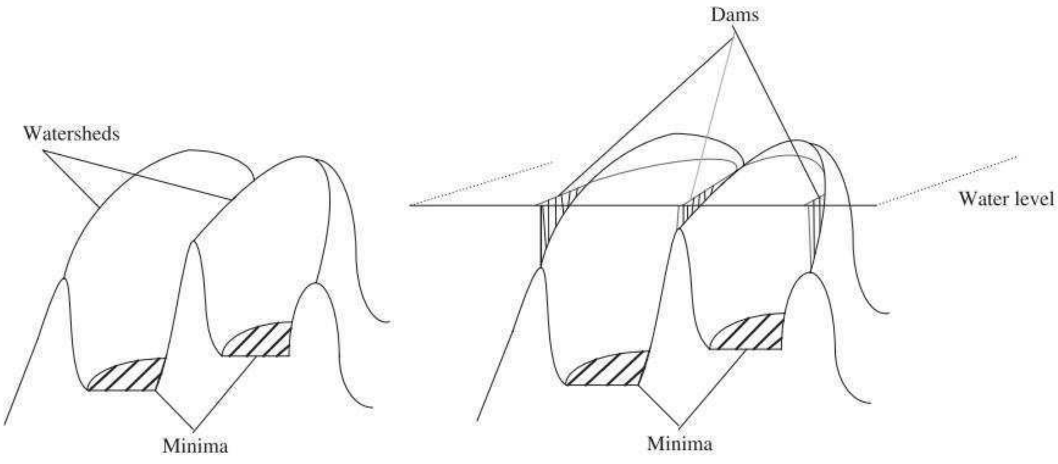
The test data comprises 42 connected component images that we call *subimages*, each containing more than one cells in an aggregate. Each has been extracted from our test set of 14 images, i.e. 3 subimages from each image, and therefore represent well the variability in the image data. The aggregation within most of the connected components also means that the simpler case of an individual cell in a subimage has not been considered, but some connected components do have non-adhering cells, hence incorporating that case in the study.

The *ground-truth* (refer to Section 3.3.2) is established by tracing out the contours of each cell in a subimage by hand. We start with a lasso that is unattached to any point in an image and anchor it to every judged salient point on the cell boundary. No interpolation is performed between any pair of these points and the lasso is non-rigid i.e. we end up with a piecewise-linear curve.

## 3.4 Applying the Watershed Transform on cellular images: the watershed algorithm

There are two broad families of watershed algorithms, following the two major watershed paradigms: the *flooding* or the *immersion* paradigm and the *rain-fall* paradigm. We briefly discuss the first in the following because it was the one initially conceptualized and techniques for markings developed for it, the latter we employ in this work.

This technique basically involves gradually immersing the surface in a water container. Previously, a hole has been made in each of the surface minima (see Fig. 3.3a). The water will begin to flow through the holes, first through those with less altitude but gradually reaching those with a greater altitude. Progressively all the catchment basins associated to the minima are flooded. The water coming from the flooding of two or more different basins might converge. At this point, suppose that a dam is built to prevent the joining. Once the whole surface is immersed, only the dams will rise above the water level, making up the watershed lines (see Fig. 3.3b). The watersheds or catchment basins are all the areas surrounded by the lines. There are several algorithms implementing this technique [19] [20], which have been subsequently improved [216] [166] [39] and even implemented in hardware [106].



(a) (b)

**Figure 3.3:** *Principles of the watershed algorithm by immersion: (a) considering a topographic image g; (b) building dams at the places where the water coming from two different minima would merge. Image courtesy Hai et. al [79].*

### 3.4.1 The Vincent and Soille algorithm

An implementation of the watershed transform by the immersion model was presented by Vincent and Soille [216]. It is a straightforward application of the idea, and could be summarized as follows:

1. *Sorting*: Compute a gray-level histogram. A list $H$ of pixels of gray-level $h$ is created.
2. *Flooding*: For all gray-levels:
    (a) Pixels having greylevel $h$ are processed. All such can be directly accessed through the list $H$.
    (b) Pixels having greylevel $h$ are potential members of catchment basin $l$ if one of their neighbors is a watershed pixel. Potential members are put in a $FIFO$ queue.
    (c) The geodesic influence zone of a catchment basin $l_i$ is the pixels having greylevel $h$ who are closer to $l_i$ than any other catchment basin $l_j$ Pixels that do not belong to a catchment basin represent new catchment basins.

Since the discussion of this algorithm in detail is beyond the scope of this work, we direct the reader to Vincent and Soille's original paper [216], and only list the one minor change, rather a simplification, that we adapt. Much of the section focuses instead on the manner in which the topographic function is defined, that is the following section, because therein lies the greater part of innovation.

#### Practical adaptation of the algorithm

The flooding of water in the image is efficiently simulated using the $FIFO$ queue of pixels. The algorithm consists of two major steps:

1. sorting the pixels *w.r.t.* increasing grey value, for direct access to pixels at a certain grey level;
2. a flooding step, proceeding level by level and starting from the minima, i.e. the three-step flooding process delineated above.

In practice, Vincent and Soille's algorithm demonstrates the undesirable property of misplaced label attribution. That algorithm could make several changes to the labels' queue during a single sweep of its neighbors, which sometimes results in incorrect labeling. This seems to be particularly a problem in higher dimensions with the correspondingly larger number of neighbours. The process to "detect and process new minima" at a level $h$ is a slightly modified version of the algorithm originally published in Vincent and Soille's paper. Here the algorithm is changed to make a sweep of the neighborhood, accumulating key information about its configuration, and then, after the neighborhood sweep is finished, make one and only one change to the labels queue. Thus we trade-off accuracy for computational speedup. However, with the classical 4− or 8−connectivity this performance degradation is insignificant against the possibility of misplacing region labels in a situation where correct labeling is paramount, such as near the boundaries lying between adjoining cells.

## 3.5 Applying the Watershed Transform on cellular images: the input data

This section exposes a two-pronged phenomenon with the watershed transform, that is, the choice of the data on which the watershed transform is applied and the strategy of that application cannot be divorced. The data dictates the strategy, for example, the use of markers (Chapter 4 Section 4.2.1) is likely to work better on a smooth topographic function image than on one with many local perturbations and therefore a tessellation incoherent with image semantics. We shall keep this in mind while discussing the topographic function and its role in the watershed transform.

In topography, the watershed line refers to a ridge that divides areas drained by different river systems while a catchment basin is the geographical area draining into a river or reservoir.

The fundamental idea leading to the watershed-based segmentation is built on this analogy. In standard image segmentation applications, contours correspond to high luminance transitions, i.e. points where the gradient norm $||\nabla f||$ takes high values. The analogy consists in regarding the gradient image function, or any other contours image such as a morphological gradient, as a *topographic relief*: the function values correspond to altitudes and the contours to crest lines of the relief, i.e. to border points of the catchment basins (Fig. 3.3).

Concurrently to the original and less pragmatic definition of the watershed transform proposed by Lantuéjoul and Beucher, a more formal definition exists in terms of skeleton by influence zone, as [20] excellently presents. The watershed line is the set of points at equal distance of the image minima, according to a certain distance: the *topographic distance.* Distance-based formulations of watershed transform are due to Meyer in the discrete case [139] and to Schmidt and Najman in the continuous case [144].

The care that must be taken with watershed segmentation is that before applying it, one must be sure whether the objects and their background do contain a regional minimum, and if crest lines outline the objects. If not, the original image must be transformed so that the contours to be calculated correspond to watershed lines, and the objects to catchment basins surrounded by them.

To this end, two image transformations have been widely studied: distance transform and gradient transform. Distance transformation is purely geometrical, and accounts for the shape of objects. However, it is only good at dealing with regular shapes, either isolated or touching objects with bottleneck-shaped connections.

### 3.5.1 The distance transform

The notion of distance is intrinsic to the watershed transform as the influence zones of regional minima are defined as the geodesic distances between them. The geodesic zone of influence $z_X(Y_i)$ of the regional minimum $Y_i$ is the set of points $X$ of the image at a finite geodesic distance from $Y$ and closer to $Y_i$ than to any other $Y_j$.

$$z_X(Y) = x \in X : d_X(X, Y_i) \text{ finite and } \forall j \neq i, d_X(X, Y_i) \leq d_X(X, Y_j) \tag{3.1}$$

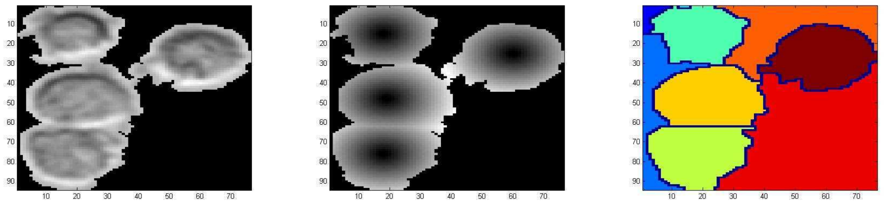
The ideal function for representing objects in the image is the *Distance Transform*, *DT*, also called *chamfer distances*; given the fact that we have achieved a primary segmentation whereby the object i.e. cell and background portions are well delimited; and because we have correlation peaks or cell centres to serve as the points of origin $P$ from which these distances will be calculated. The measure of distance is therefore

$$DT(P)[X] = \min_{Y \in P} \text{ dist(X, Y)} \tag{3.2}$$

for each point $x$ in the image to the nearest point $y$ in $P$. The calculation of chamfer distances on binary images is performed in practice by propagating local distances as min-sums forward and backward passes as in the popular $SLT$ algorithm by Rosenfeld and Pfaltz [173]. The distance used is the Eucledian distance i.e. $\text{dist}(X, Y) = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2}$ for the $2D$ dimensions 1, 2.

Watershed segmentation applied to distance transformed binary images (usually binarised through thresholding) is useful for separating touching objects that are convex (see [130, 46, 220]). This means that on the distance map, all the pixels of any connected component will have for intensity their distance to the closest pixel belonging to the corresponding centre. But since this would simply produce a Voronoi tesselation, we hope that the separation calculated by the watershed lies on the real boundaries that separate two cells.

Once the peaks in the correlation space are located (Chapter 2 Section 2.3.3), they are ideal candidates to serve as the set of markers for the subsequent watershed (e.g. in Fig. 3.4 these centres are initial propagation points for the distance transform) because of the fact that for any given cell its corresponding peak is always found to lie within the cell. That is the reason why we variously call these peaks the centres of our cells, even though they are only pseudo-centres.



(a) (b) (c)

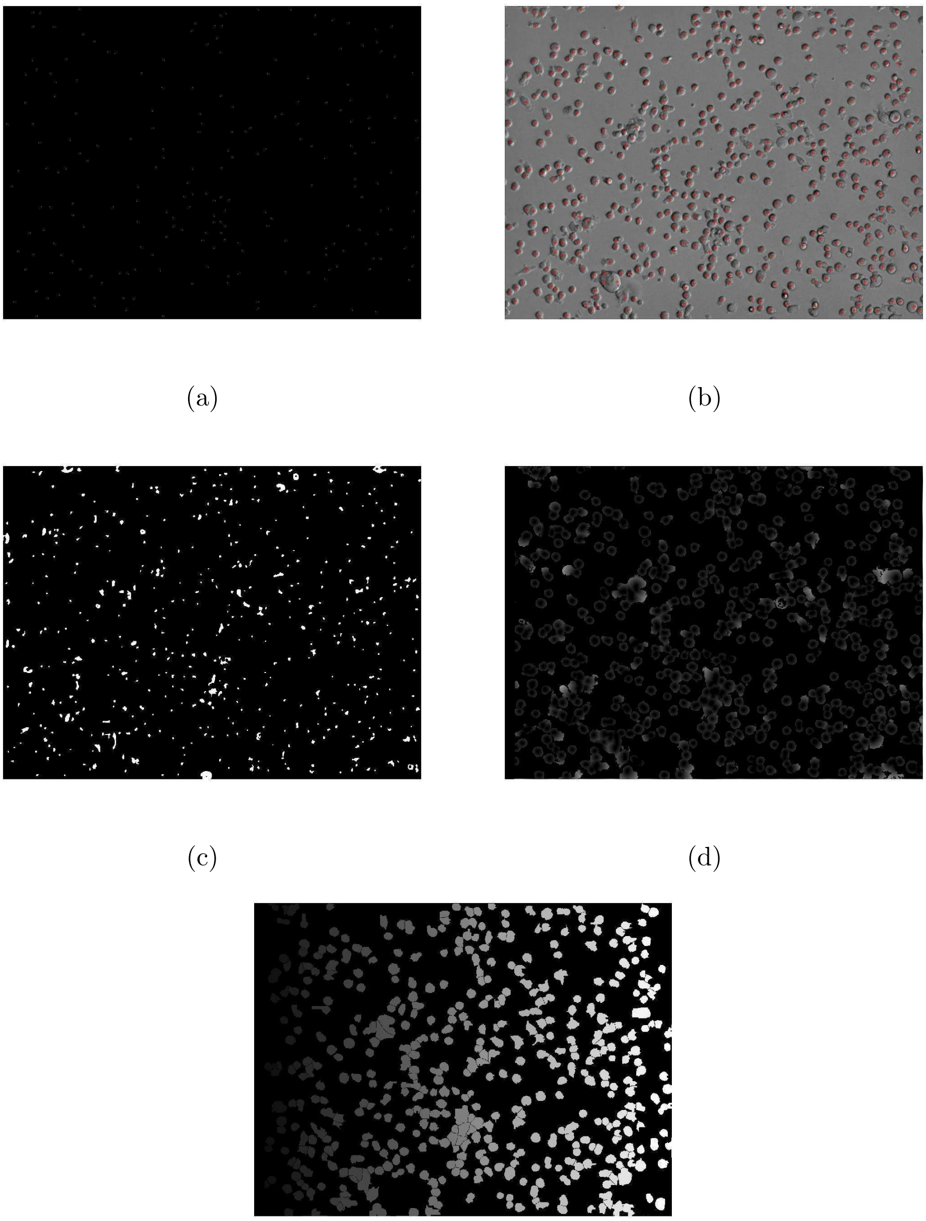
**Figure 3.4:** *(a) A cell cluster, (b) the corresponding chamfer distance map with correlation peaks for the distance transform reference points, and (c) the corresponding watershed.*

The peaks do constitute our set of markers for any subsequent mathematical morphological operation but we try to augment it with an *a priori* pertaining to the morphology of the cells around them. For this purpose we globally threshold our Chamfer distance map to obtain an extended maximum around the peaks. The assumption is that the central part of the chamfer "hill" around each peak mimics the overall shape of the "hill", and therefore the cell. This assumption is empirically observed to hold to the extent that the extended maxima are round-ish for round cells and longer for long ones, which is all very well, but may have significantly different shapes to their parent cells or touch each other in an agglomerate. Fig. 3.5 shows the images involved in segmentation.

We notice that watershed segmentation based on chamfer distance transform does not necessarily

1) segment cells right at their cell walls, and

2) does not necessarily correctly decompose a connected component; instead decomposing them in an equitable manner based on the chamfer distance, even though we introduced *a prior* information in the form of extended maxima.



(a)

(b)

(c)

(d)

(e)

**Figure 3.5:** *(a) Peaks or peudo-centres, (b) superimposed on the original image, (c) their extended maxima as initial distance transform fronts, (d) the chamfer distance map from markers to connected component edges and (e) the labels given by the watersheds.*

### 3.5.2 The gradient-weighted distance transform

Contrary to the geometric distance transform, the gradient transformation is intensity-based, assuming that inter-cellular gradients are higher than intra-cellular ones gradients. As with all gradient-based operations, this transformation is sensitive to imaging noise, and usually results in over-segmentation.

To overcome the above difficulties, namely non-realistic segmentation of the distance transform and the extreme over-segmentation of the gradient transform, we use a combined image transformation called the "gradient-weighted distance transform" or $GWDT$, which accounts for both geometric and intensity features. The distance element smooths out the many local minima generated by the gradient, by augmenting it with a monotonically-increasing component, and the gradient component adds critical boundary information for better placement of watershed boundaries.

The geometric distance transform $D$ and the gradient transform $G$ must be combined into a single representation that captures the object separation cues available in the data. One challenge in this regard is the fact that these quantities are dissimilar, i.e., they are expressed in different units, and they can be normalized differently. The final result of the combining operation should be in distance units. These conflicting requirements are met by the following formula [119].

$$D_w = D \times \exp(1 - \frac{G - G_{min}}{G_{max} - G_{min}}) \tag{3.3}$$

where $G_{min}$ and $G_{max}$ are the minimum and maximum values of the gradient $G$ (the Deriche gradient [49] is used) needed for normalization. Note that the distance value $D_w$ is high at positions closer to the center of foreground objects, and in pixels with smaller gradient values. $D_w$ is smaller close to the boundary of the foreground objects, or where the gradient is relatively large. Intuitively, this captures the essential object separation cue that pixels with bigger gradient values tend to be on the boundary of an isolated object, or on the boundary between two touching objects.

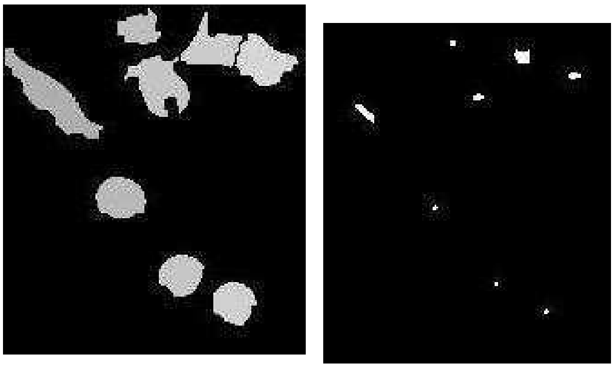
### 3.5.3 Building cell shape priors into the distance map

Usually only a marker's presence is used in the watershed algorithm, and marker structure remains unemployed. This section presents a method to increase the performance of marker-guided watershed by using the information related to the markers' placement, shape and size. Such *a priori* information is commonly exploited in other segmentation schemes such as active contours [113]. It is important to note that the term *marker* is being used generally as a small connected component that helps regulate an operation on a larger one, not necessarily as watershed markers.

It is preferable that prior shape information be available in markers since it would affect object shape irrespective of the image function. This means for instance that the watershed algorithm will degrade gracefully in case of use of a simple geodesic distance map.

Pixel-markers would work well if all cells have the same size and shape or if the image function has clearly demarcated object boundaries. But since that is rarely the case, point-markers on a distance map could bias watershed segmentation results for cells lying in clusters. For this purpose we would like the regional minima around the discovered peaks to imitate the shape of the cell they represent. The halos image candidates perfectly as the date-source for these regional maxima, since it presents a relief that elevates toward each peak, in various manners as we have previously seen depending on the shape of the cell. We notice that the elevation around peaks of circular cells are more pointy as well as circular, while those of spread cells they form particular longitudinally-concave crests. Thus the use of regional maxima around centres as watershed markers will allow shape information to be propagated over the distance map.

As a marker is always contained within object boundaries, we devise a method to detect approximate cell boundaries by a point-marker guided watershed, and then work within each cell region to construct the marker. Thus in this *windowed h-home* the value of $h$ is dynamically calculated within the region representing the cell. The local $h$ is therefore the Otsu threshold of this region. Fig. 3.6b) shows an example of the markers thus produced. They compare favorably with the parent cell in Fig. 3.6a.



(a) (b)

**Figure 3.6:** *Shape relationship between (a) cells and (b) corresponding watershed markers.*

The problems however that exist with these methods include:

- the frequent aberrant cases of cells where the gradient information is not strong enough to stop the flooding process, and the watershed thus produced reflects simply the distance map, cutting cells in arbitrary shapes.
- malplaced centres: cell centres detected multiple times on the same cell force the cell to be separated into more than one segments even if left naturally without markers the cell would have formed a unique basin.
- the marker might be mal-formed, because the process leading up to its formation had previous errors, making the segmentation incoherent with the ground-truth

### 3.5.4 Partial membership probabilities as the topographic function

As mentioned in the beginning of the section, we were looking for a topographic function that is independent of extraneous influences like the distance map, and more robust to the placement of watershed markers or even their absence.

Let us recall that one of the main problems in image segmentation is uncertainty. Some of the sources of this uncertainty include additive and non-additive noise, imprecision in computations and vagueness in class definitions. Traditionally, probability theory was the primary mathematical model used to deal with uncertainty problems; however, the possibility concept introduced by the fuzzy set theory has gained popularity in modeling and propagating uncertainty in imaging applications. Let us succinctly explain the *Fuzzy C-Means Clustering* algorithm and then how the algorithm is adopted for this work.

In hard clustering, data is divided into distinct clusters, where each data element belongs to exactly one cluster. In fuzzy clustering (also referred to as soft clustering), data elements can belong to more than one cluster, and associated to each element is a set of membership levels. Refer to Dunn [54] for a detailed treatment of the subject. These indicate the strength of the association between that data element and a particular cluster. Fuzzy clustering is a process of assigning these membership levels, and then using them to assign data elements to one or more clusters. One of the most widely used fuzzy clustering algorithms is the $Fuzzy\ C - Means$ $(FCM)$ $Algorithm$ [21]. The FCM algorithm attempts to partition a finite collection of $n$ elements $X = x_1, ..., x_n$ into a collection of c fuzzy clusters with respect to some given criterion. Given a finite set of data, the algorithm returns a list of $c$ cluster centres $C = c_1, ..., c_c$ and a partition matrix $U = u_{i,j} \in [0, 1],\ i = 1, ..., n,\ j = 1, ..., c,$ where each element $u_{ij}$ tells the degree to which element $x_i$ belongs to cluster $c_j$ . Like the $k$-means algorithm, the $FCM$ aims to minimize an objective function.

In fuzzy clustering, each point has a degree of belonging to clusters, as in fuzzy logic [233], rather than belonging completely to just one cluster. Thus, points on the edge of a cluster, may be in the cluster to a lesser degree than points in the center of cluster. For each point $x$ we have a coefficient giving the degree of being in the $k$th cluster $u_k(x)$. Usually, the sum of those coefficients for any given $x$ is defined to be 1:

$$\forall x \left( \sum_{k=1}^{\mathrm{c}} u_k(x) \ = 1 \right). \tag{3.4}$$

With fuzzy $c$-means, the centroid of a cluster is the mean of all points, weighted by their degree of belonging to the cluster:

$$\mathrm{center}_k = \frac{\sum_x u_k(x) x}{\sum_x u_k(x)}. \tag{3.5}$$

The degree of belonging is related to the inverse of the distance to the cluster center:

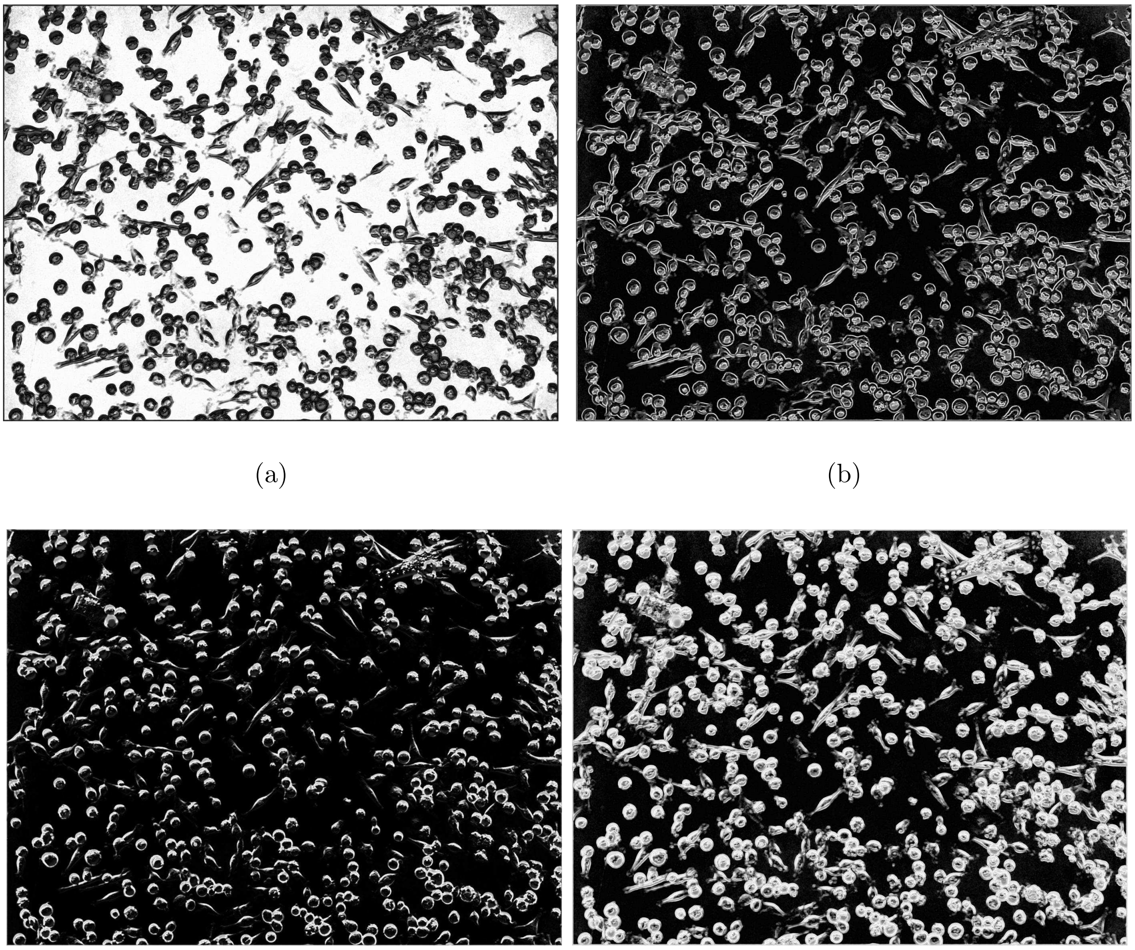
$$u_k(x) = \frac{1}{d(\mathrm{center}_k, x)}, \tag{3.6}$$

The fuzzy $c$-means algorithm is very similar to the k-means algorithm:

- Choose a number of clusters.
- Assign randomly to each point coefficients for being in the clusters.
- Repeat until the algorithm has converged (that is, the coefficients' change between two iterations is no more than a sensitivity threshold $\varepsilon$):
  - Compute the centroid for each cluster, using the formula above.
  - For each point, compute its coefficients of being in the clusters, using the formula above.

The algorithm can be regarded as a variation on $k$-means but with partial membership in classes.

In spite of a slew of modifications available for the watershed transform, little interest has been shown to using a pre-classified topographic image. Of course, hard classification is final in its decision, and a soft method such as the one described above is required. The sole authors found to have studied this possibility are [50], where pixels are represented by the memberships to each class of interest. A morphological gradient is computed from this representation on which watershed is applied.



(a) (b)

(c) (d)

**Figure 3.7:** *Fuzzy-C-means class assignment to (a) background (b) cell matter and (c) highly-contrasted cell matter or walls. (d) represents the sum of the scores in (b) and (c) i.e. the net probability of a pixel belonging to the class cell.*

Using a very similar idea, we assign one of the three membership-classes to each pixel in the image: *background*, *cell-inside* and *cell-borders*. *cell-borders* is somewhat of a misnomer because its higher membership degrees represent highly-textured parts of the cells as well as those near and around its cell walls. However, the manner these degrees are used (post-classification re-combination) renders the difference insignificant. Combined, *cell-inside* and *cell-borders* represent the cell pixels in the image. The sum of the labels in *cell-inside* and *cell-borders* represents our topographic image. The desirable property of this image is that all regional minima among the cell pixels are higher than their counterparts in the background (yet we still do not have a connected component vs background decision). This means, used in a hierarchical watershed scheme, the basins located in cells will become extinct later than those in the background, as well as more slowly than those in the background. Refer to the section 4.2.2.A on hierarchical watersheds.

### 3.5.5 Comparison and Discussion

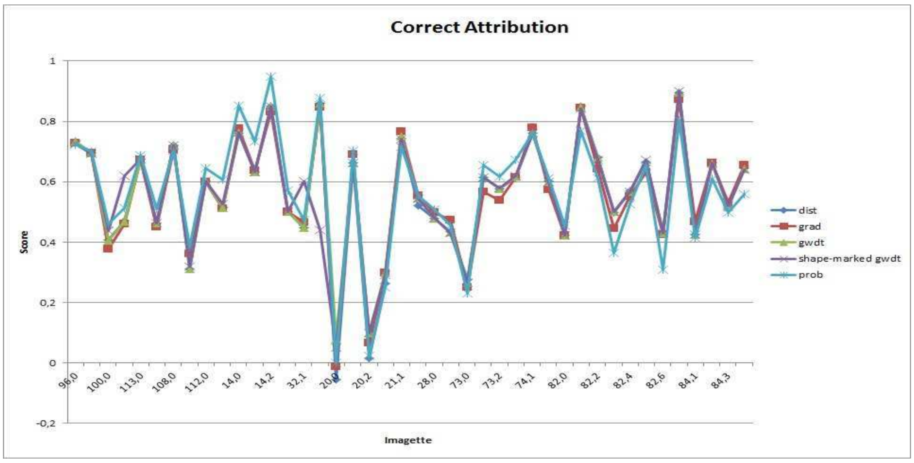
The nature of the topographic image function always points toward the desirable properties of higher values close to object boundaries but its constitution could very well vary, as we have shown in the preceding subsections. Obviously, the first function i.e. simple chamfer distances is insufficient for images where cuts following finely the object boundaries are necessary. Even so, we will include it in our analysis. The other topographic data are either currently used in medical image processing or have been proposed as in subsection 3.5.4. In any case, this part of the section will aim to benchmark the five functions mentioned on the same test data sample so that for the latter part of the work could be carried out on the one that is best suitable for our purposes.

Each subimage is then loaded and undergoes a segmentation using each of the five topographic functions, and a geodesic-reconstruction-by-dilation marker (Annex A). This marker is needed for the case where the segmentation fragments the cell the fragment geodesic-dilation-marked by the centre is identified as the cell, since at this stage no post-segmentation-merging is applied. At each turn, it is then a simple matter to compare each resulting cell-object and the ground-truth according to the discrepancy criterion in 3.3.2.

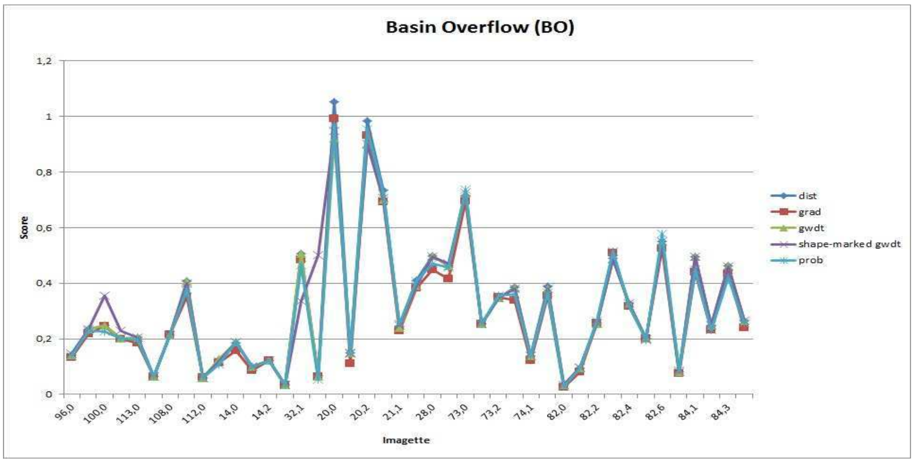
The resulting data thus compiled are too copious to reproduce raw, instead we have graphed them for a more efficient spotting of trends within them.Figures 3.8 through 3.9 illustrate the comparisons.

Fig. 3.8 compares (a) correct attribution ($CA$), (b) basin overflow ($BO$) and (c) basin shortfall ($BS$) for the subimage set. Several trends is noticeable, and it is only fair to discuss them individually and at proper length, as in the following:

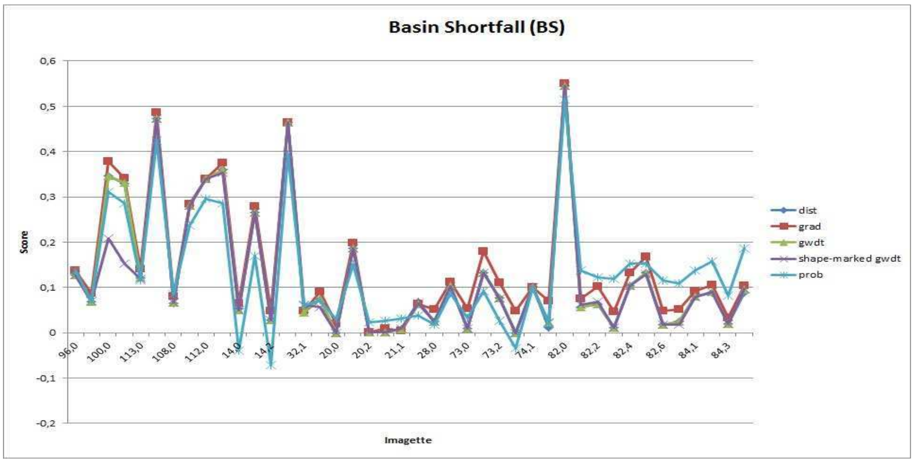
- The subimages present a diverse test data set, and we thus obtain non-flat curves with singular corners.
- All curves follow very similar trajectories, since all data points lie in tight neighborhoods. The advantage of one topographical function over the others



(a)



(b)



(c)

**Figure 3.8:** *Graphs of (a) Correct segmentation (b) basin overflow and (c) basin shortfall for the entire subimage data.*

is therefore marginal, except for the fuzzy probability function.

- The simple chamfer distance does not fare too badly considering its relative lack complexity of calculation when compared to the shape-guided gradient-weighted chamfer distance. The first two and the fourth curves practically overlay each other.

- The gradient plays the maverick in (a) and (b), due to its different BS. This could be attributable to the rather brusque nature of the gradient image, which has not been smoothed or post-processed in any way, which causes the basins to form inside of the high-gradient band representing cell walls, thus causing the high basin shortfall values and lowering those of correct attribution. But at the same time the gradient also demonstrates lower basin overflow, something that reaffirms the basins-forming-inside hypotheses, which is advantageous for a segmentation in which deeper basins risk spilling over into shallower ones, as can happen in case of the simple chamfer topographic function if a marker were not used.

- The sum-of-fuzzy-probabilities-function demonstrates the highest correct attribution for the majority of subimages, and the lowest basin shortfall again for the majority of the subimages. This could be attributed to the within-cell and near-cell-wall probabilities being very distinct from the background probabilities. The actual gain in precision is small but non-negligible.

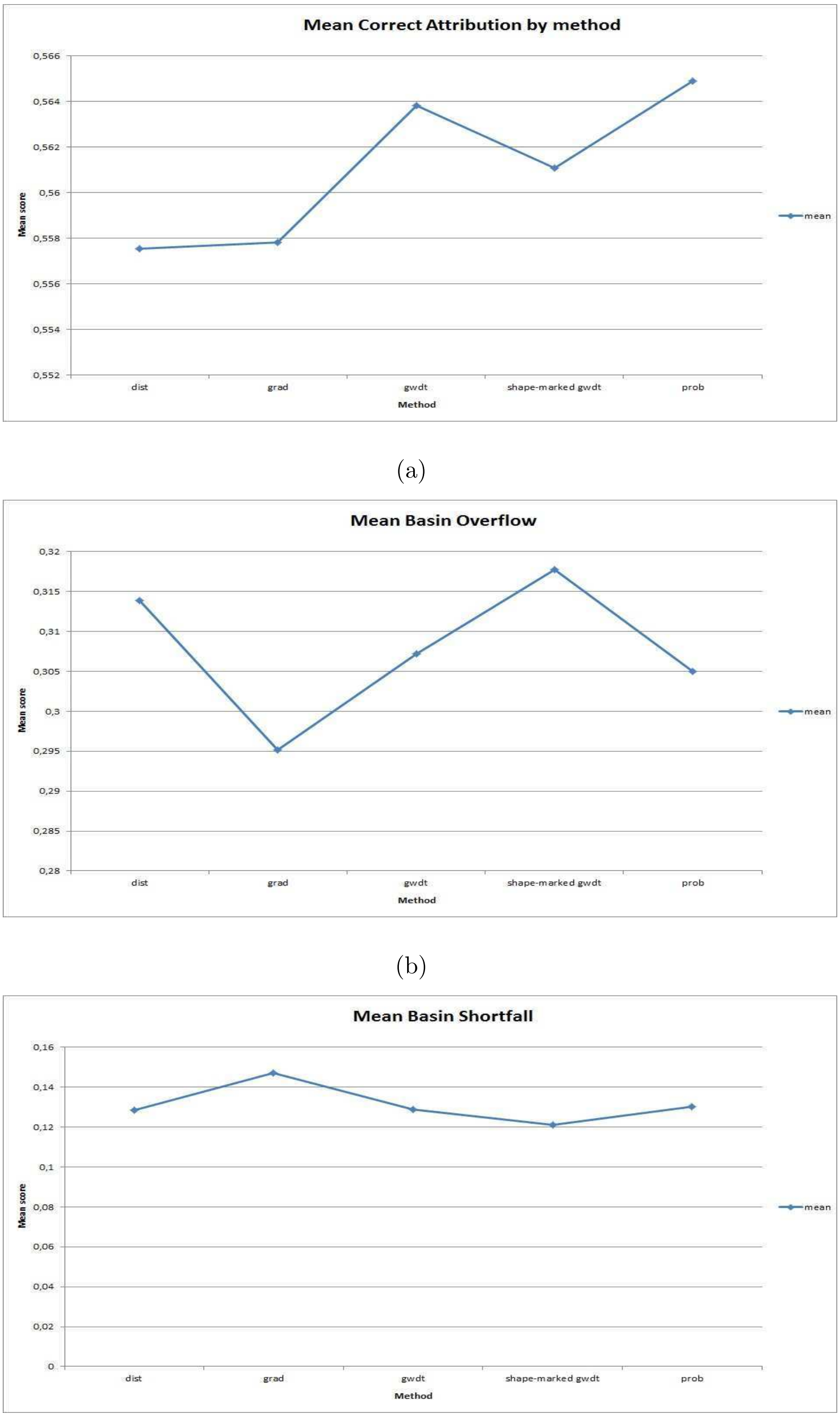
- The shape-guided $GWDT$ does not fare too well because of the issues with the gradient used in its construction, and also because of the possibly-imperfect shaping of the $DT$ initial propagation fronts.

**Table 3.1:** *Mean and standard deviation of Correct Attribution by topographic function.*

| method | dist | grad | gwdt | shape-gwdt | prob |
|---:|---:|---:|---:|---:|---:|
| mean | 55.75% | 55.78% | 56.38% | 56.11% | 56.49% |
| stddev | 20.28% | 19.30% | 18.77% | 18.25% | 20.35% |

Fig. 3.9 and Table 3.1 illustrates the trend of the mean values of the segmentation quality parameter scores, in the same order as the previous image:

- All methods have on the average scored less than 70% according to our correct segmentation criteron.



(a)

(b)

(c)

**Figure 3.9:** *Trends in mean of (a) Correct segmentation (b) basin overflow and (c) basin shortfall.*

- The gradient scores the poorest on $CA$ but the highest (lowest value) in terms of $BO$. Barring the gradient from the list, we would have flat $CA$ and $BS$ curves, hovering around 0.65% and 0.07% respectively.
- The fuzzy probability function shows a slightly superior $CA$ over gradient, but also reasonably superior $BO$ and a $BS$ that is only slightly worse off compared to all other topographic functions involving a distance measure.

**Table 3.2:** *Mean and standard deviation of Basin Overflow by topographic function.*

| method | dist | grad | gwdt | shape-gwdt | prob |
|---:|---:|---:|---:|---:|---:|
| mean | 31.38% | 29.51% | 30.72% | 31.77% | 30.50% |
| stddev | 23.92% | 22.73% | 22.27% | 22.10% | 23.07% |

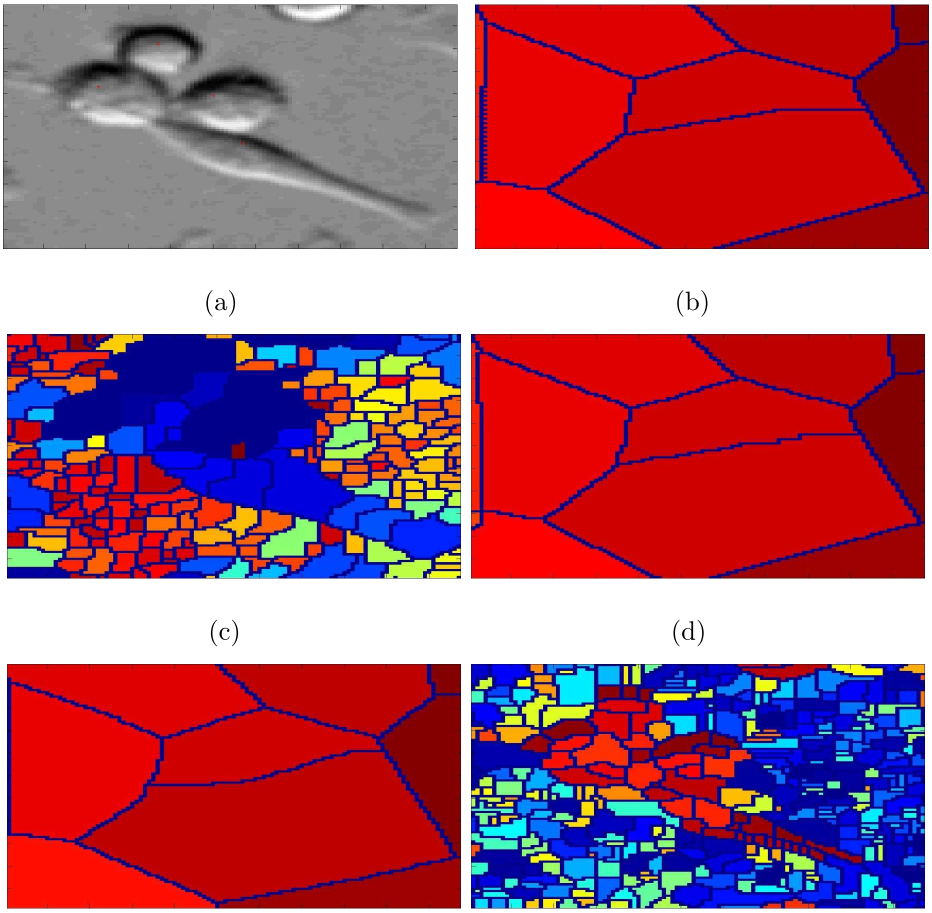
**Table 3.3:** *Mean and standard deviation of Basin Shortfall by topographic function.*

| method | dist | grad | gwdt | shape-gwdt | prob |
|---:|---:|---:|---:|---:|---:|
| mean | 12.86% | 14.71% | 12.89% | 12.12% | 13.01% |
| stddev | 14.49% | 14.10% | 14.47% | 13.70% | 12.54% |

### 3.5.6 Conclusions and opening up to following work

The previous section not only exposes various topographic functions that one could use in order to achieve desirable partitioning of an image, but it also exposes how ill-adapted some of them are for applying to our image data. In fact, in this section we wish to re-visit that discussion, and have saved for this point the most discriminating results, i.e., a visual inspection. One must bear in mind that the use of, for instance a *DT* map, as the topographic relief for the watershed transform is a complete image segmentation strategy, as authors have shown [71, 24, 20, 72, 219, 130, 183], and much more consequential than simply having built an image representation. The segmentation over any topographic function not only defines the regions that are assigned logically to objects in the image, but also defines the shape and extent of those regions, and sometimes dictates the post-segmentation corrective process as well [3, 64].

To illustrate this with examples, let us invite the reader's attention to the segmentations produced in the previous section, as shown in Fig. 3.10. The distance transform (Fig. 3.10 b. is simply a Voronoi tessellation and completely disregards the shape of the cells. It has managed to separate each cell centre (obtained in Chapter 2) by perpendicular bisectors equidistant from each centre. Weighting by



(a) (b)

(c) (d)

(e) (f)

**Figure 3.10:** *Segmentation of the subimage a)* 0032.1*, using for topographic relief function: b) chamfer distance map, c) Deriche gradient, d) GWDT, e) shape-guided GWDT, and f) fuzzy C-means probability map.*

the gradient (Fig. 3.10 d.) does little to alter the original tessellation albeit in fine details, and distance weights calculated from different-sized centre-grains (Fig. 3.10 e.) does a little better in respecting object-region sizes and shapes, e.g. the spread cell region encroaches its neighbouring cells less, and region boundaries are more convex, following the convex cell shapes, rather than straight lines.

Of course, after this segmentation, the segments are restricted to within the agglomerate connected components of the binarised image, thereby providing us with regions representing individual cells within the *ACC*s. But the original inaccuracy of the segmentation produces regions that do not accurately represent cell shapes and sizes. For many segmentation applications in the literature [12, 167, 77, 130, 220] this does not pose a problem, but since our goal is to be able to distinguish different classes of metastasic cells, this lack of segmentation accuracy renders the following classification difficult and impinges on its accuracy as well.

Topographic functions without external information imposed on them, such as the gradient (Fig. 3.10 c.) and the probability maps (Fig. 3.10 f.) however, are much more realistic in the manner in which they match the segmented regions and real object boundaries. This is desirable to our application for the reason presented in the previous paragraph, although these methods introduce a different problem: *over-segmentation* (treated in detail in following chapter's section 4.1). But before passing on to that subject, let us for a further moment adhere to the analysis of the figure. The gradient map (Fig. 3.10 c.) produces shallower basins than the probability map (Fig. 3.10 f.), and these could be relatively easily confused by a corrective algorithm with the shallow background basins. And if we looked further, the former are also larger and exceed the cell contours and into the background, and the latter do not. This reinforces the decision arrived at in Section 3.5.5 to use the probability map as the topographic function.

The use of the probability map in this manner only offered us a correct segmentation score of around 56% according to our discrepancy criterion. As an outcome of this comparison it seems logical to select the fuzzy-probability map over competing topographic functions. There are other reasons for this choice as well. In a hierarchic watershed scheme, as we shall see later (Chapter 4 Section 4.2.2.A) the probability function attenuates much slower than its gradient counterpart, resulting in a much more precise hierarchy. Also, our qualitative-only eyesight criterion which complements our discrepancy criterion for segmentation quality evaluation clearly indicates its superiority over the chamfer-distance-only function, the latter being able to only make straight-line cuts in the image and failing to follow the curve of the cell wall. Thus the probability function will constitute our topographic functions of choice for the ensuing segmentation. And because of the over-segmentation, the regions do not correspond to cells in the images. Therefore we were compelled to look at strategies to improve the segmentation thus obtained by mitigating the effects of over-segmentation, and this aspect was investigated in an amount of detail that necessitates its presentation in a separate chapter. Therefore in the following chapter we investigate two major axes of methods used to mitigate the problem of over-segmentation inherent in the use of the probability map as the topographic function before finally arriving at the solution we propose for this application in particular.

# 4

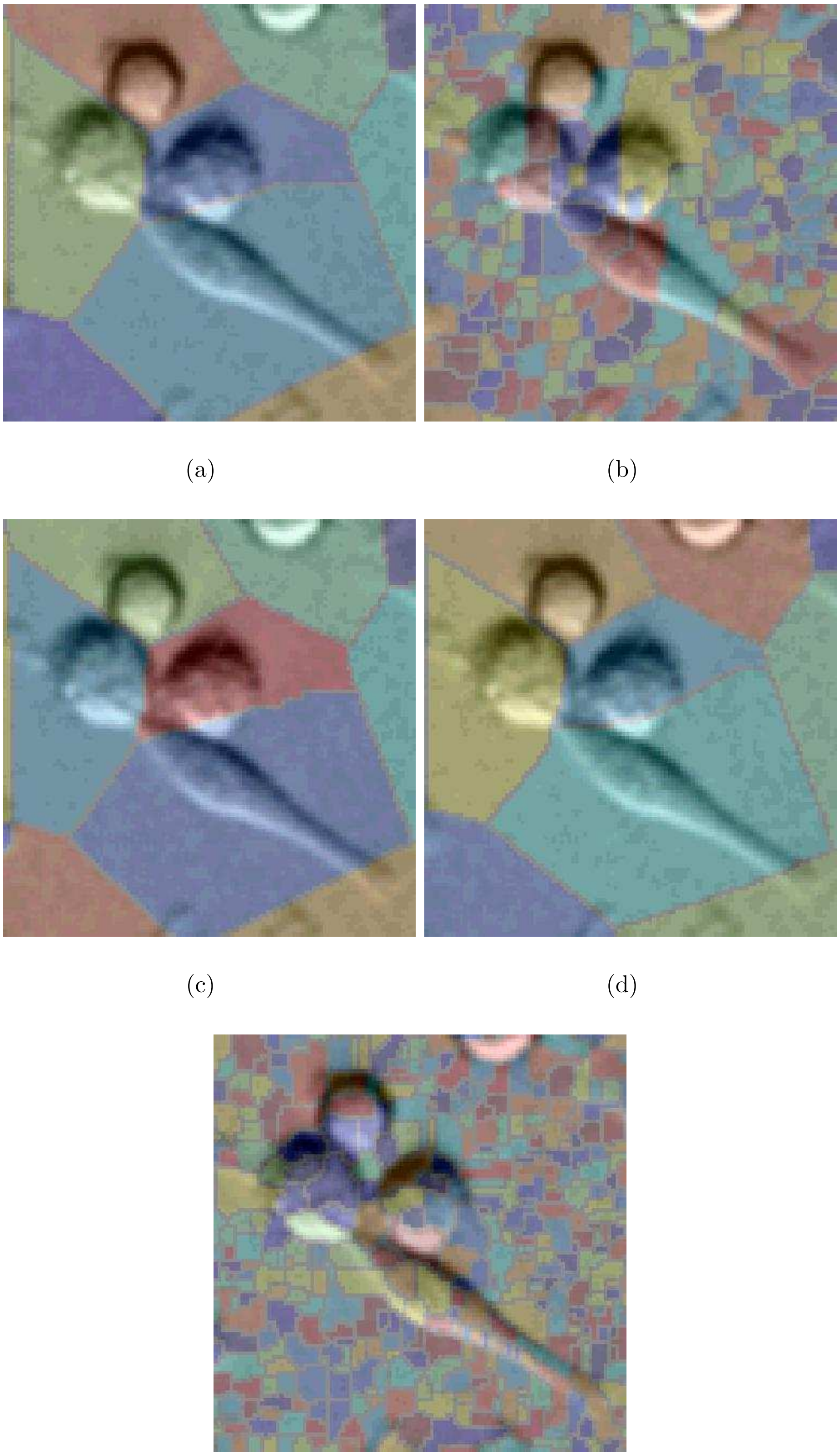
# Improving the segmentation

The previous chapter presented implementations of various watershed segmentation application algorithms utilizing different topographic functions that one could use in order to achieve desirable partitioning of an image, and it concludes with the choice of the fuzzy probabilities map as the most appropriate one for applying to our image data. We build this chapter up by continuing with the same discourse and re-visiting the same example as the concluding figure offered. The application of the watershed transform on the probability map without the use of any segmentation-saliency aids such as marking or modifying the topographic function did produce watershed regions that adhered well to cell boundaries and were well distinct from the rest of the image, but were many and small and incoherent with the desired object shapes. We begin by looking at this familiar problem called over-segmentation. We then offer some solutions for this problem, including four implementations inspired from the existing literature as well as the one we conceived, compare them, and include the best-performing one into our processing chain that emits bounding boxes of the segmented cells, called imagettes, to the final phase of cell classification.

## 4.1 The problem of over-segmentation and resolution strategies

Let us once more invite the reader's attention to the segmentations produced in the previous chapter, now shown superimposed over the original subimage for a better appreciation in Fig. 4.1. We surmise as before that topographic functions without external information imposed on them, such as the gradient (Fig. 4.1 b.) and the probability maps (Fig. 4.1 e.) however, are much more realistic in the manner in which they match the segmented regions and real object boundaries. The gradient map (Fig. 4.1 b.) produces shallower basins than the probability map (Fig. 4.1 e.), and shallower basins being less distinguished from those in the background and are therefore difficult to correctly unify into cell objects through the use of corrective post-segmentation-processing as described in Section 4.4. And if we looked further, the former are also larger and exceed the cell contours and into the background, and the latter do not. This reinforces the decision arrived at in Section 3.5.5 to use the probability map as the topographic function.

The above appraisal serves well as an exponent of the inherent problem with the watershed, that it often results in severe over-segmentation, even if appropriate filters are used for the original image or for the topographic gradient image before the watershed operation is performed. This is due to the fact that the *topographic image*



(a) (b)

(c) (d)

(e)

**Figure 4.1:** *Segmentations from Fig 3.10 in the described order, this time superimposed on the subimage.*

over which the watershed transform is calculated, usually the gradient, exhibits too many minima. The coarse application of watershed is due to the fact that each minumum gives rise to a catchment basin. Many minima are produced by small variations, mainly due to noise, in the grey values.

This over-segmentation could be reduced by appropriate filtering, such as averaging and smoothing. However, all the catchment basins do not have the same importance. Some of them are induced by noise, others are minor structures in the image. Hence, techniques for reducing over-segmentation have been developed that take into account the saliency or the relevance of a catchment basin to the particular watershed application, either automatically or through human input. These techniques fall into two categories: **preventative** and **corrective**. In the following sections we will review some of these techniques before presenting our own contribution to this solution.

## 4.2 Preventing over-segmentation

As mentioned before, without any preprocessing, the number of regions extracted equals the number of regional minima of the topographic image function used and these are often extremely numerous leading to an over-segmentation. For this reason, the watershed is generally computed from a modified topographic function that allows us to privilege certain and much fewer minima or indeed allow them exclusivity as sources to the flooding process. In the case where the sources for the flooding are not all minima of the topographic image, two solutions are possible:

1. First, use the *markers* (Section 4.2.1) as sources: in this case, catchment basins without sources are flooded from already flooded neighboring regions.

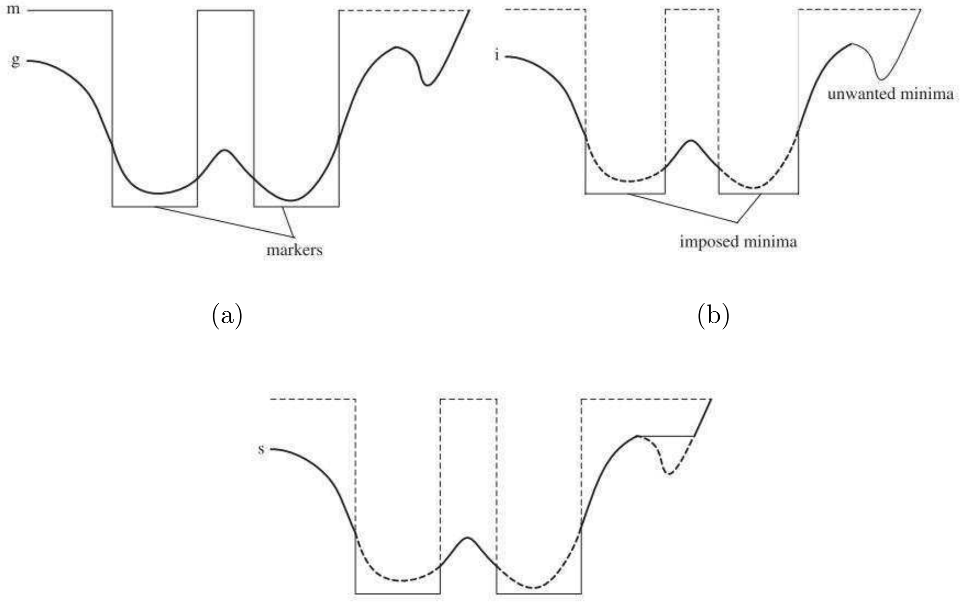
2. The second solution consists in modifying the topographic surface, such as to change its homotopy, in such a way that the desired or significant minima become the only regional minima. The most well-known homotopy modification operation is called *swamping* (Section 4.2.2).

Both the methods mentioned above make use of morphological image reconstruction or geodesic erosion that is explored in more detail in Annex $B$ . The following two subsections are dedicated to the examination of the two mechanisms of preventing over-segmentation.

### 4.2.1 Selecting desired minima through Marking

To avoid over-segmentation, a very powerful method was introduced by Meyer [140]. Suppose we know *a priori* that a connected set of points belong to an object, and such a connected set for each object to segment, called *markers* in the literature. If we could modify the image on which to compute the watershed so as to impose these sets as regional minima, we can then obtain a watershed that has a region around each object, since each catchment basin represents either the background or one unique object.

Meyer proposed a direct marker-based watershed algorithm by immersion. In this algorithm, the flooding process is performed directly on the marker-modified gradient image $i$ instead of the final modified gradient image $s$ as shown in Fig. 4.2, and unwanted minima (and consequently, unwanted regions) are suppressed during the algorithm itself.



(a) (b)

(c)

**Figure 4.2:** *1D illustration of marker-induced topographic image modification: (a) the topographic (gradient) image $g$ and the marker image $m$; (b) imposing markers on the topographic image $g$ as minima and get the marker-modified topographic image $i$; (c) suppressing all unwanted minima and get the final modified topographic image $s$. Image courtesy Hai et. al [79].*

The marking approach is applied as follows: first, we define the properties which will be used to mark the objects. These markers are called *object markers*. The same is done for the background, i.e., for portions of the image in which we are sure there is no pixel belonging to any object. These markers constitute the *background markers*. The rest of the procedure is straightforward and is the same for all applications: the gradient image is modified in order to keep only the most significant contours in the areas of interest between the markers. This gradient modification consists in changing the homotopy of the function. Then, we perform the final contour search on the modified gradient image by using the watershed transformation. No supervision, no parameter and no heuristics is needed to perform the final segmentation. The parameterization controlling the segmentation is concentrated in the marker construction step where it is easier to control and validate it.

The technique of marker-controlled watershed allows us to look for the contour of the objects with less exactitude and guarantees the number of contours found, i.e. one for each object. All the difficulty lies in determining the markers, i.e., to a localization of the objects.

In brief, a segmentation by marker-controlled watershed therefore constitutes of the following steps:

1. Find the markers, one for each object and one (or more) for the background
2. Compute the topographic function, usually a contrast image e.g. the gradient image
3. Impose the minima of the markers by greyscale geodesic reconstruction
4. Compute the watershed.

### 4.2.2 Eliminating non-salient basins through Swamping

The second solution consists in modifying the topographic surface in such a way that the markers become its only regional minima. This operation is performed using greyscale image reconstruction by erosion, an operation which modifies the minina dynamics of the image. Let us examine it briefly.

**Minima dynamics** The *dynamics* [73] of a regional minimum is a criterion of contrast. If we recall, a regional minimum is a connected set from which it is impossible to reach a point with a lower height without having climbed. The minimum height of this climb is the valuation of the contrast of the regional minimum.

The concept of dynamics is illustrated in Fig. 4.3. It can be used, as we shall shortly see, to find relevant regional minima by geodesic reconstruction of a topographical function. In practice, we do not impose these minima as markers by geodesic reconstruction of a marked function. On the contrary, we suppress the regional minima of $f$ with a dynamics lower than a given contrast value $h$. The standard algorithm to achieve this relies on the computation of the geodesic reconstruction by erosion $E_f^{\infty}(f_h)$ of $(f_h)$ over $f$ where $(f_h)(a) = f(a) + h$.

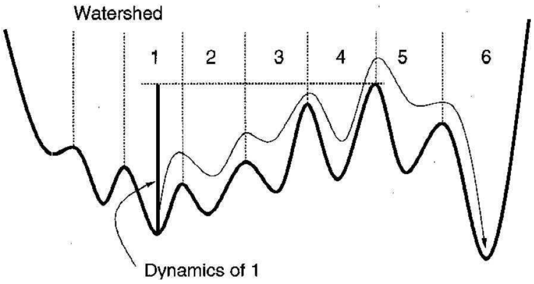


**Figure 4.3:** *Illustration of the concept of dynamics.*

The mechanism for morphologically reconstructing by erosion an image at any level $h$ is illustrated in Fig. 4.4 which contrasts with the previous in the number and the height of catchment basins which will engender segmented regions when the watershed transform is applied on the image. Only the basins having a depth greater than the swamping level $h$ remain after the reconstruction step. This alters the dynamics of each catchment basin but also the number of regional minima, hence homotopy modification.

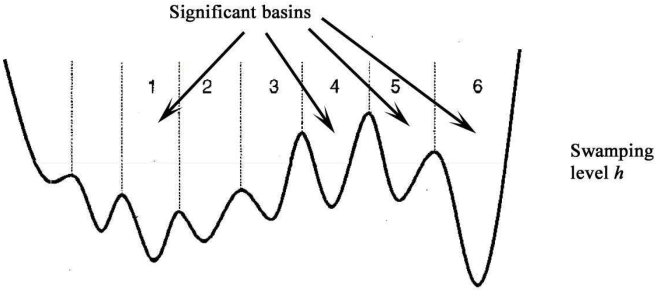


**Figure 4.4:** *Topographic function geodesically eroded by waterfall swamping until only significant basins remain.*

### 4.2.2.A Watershed segmentation hierarchies and the Waterfall algorithm

Thus far, we have seen as a means to prevent over-segmentation, marking and using homotopy modification, to produce as many catchment basins as there are objects in the image. This section presents *hierarchical watershed segmentation* as an alternative. Originally developed by Beucher [17], this approach, rather than preventing over-segmentation, computes the importance of watersheds with respect to given criteria.

First let us mathematically define what is meant by a hierarchy.

**Definition** *Let* $P_{h_i}$ *be a sequence of partitions of the plane. The family* $(P_{h_i})_i$ *is called a* ***hierarchy*** *if* $h_i \geq h_j$ *implies* $P_{h_j} \subseteq P_{h_i}$*, i.e., any region of partition* $P_{h_i}$ *is a disjoint union of regions of partition* $P_{h_j}$*.*

There exist many methods for building a hierarchical segmentation [156], which can be divided in three classes: *bottom-up*, *top-down* or *split-and-merge*. A recent review of some of those approaches can be found in [189]. A useful representation of hierarchical segmentations was introduced in [145] under the name of saliency map. This representation has been used (under several names) by several authors, for example for visualisation purposes [75] or for comparing hierarchies [5].

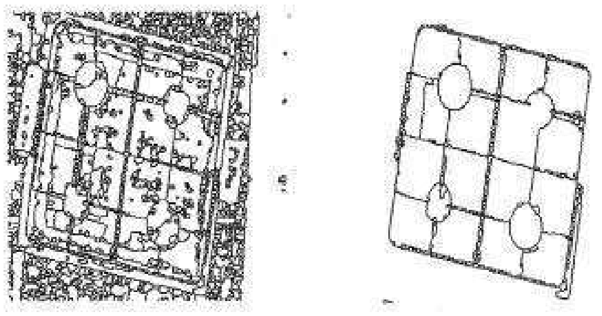
For bottom-up approaches, a generic way to build a hierarchical segmentation is to start from an initial segmentation and progressively merge regions together [155].

Every hierarchy can be assigned a *saliency map*, by valuating (i.e. assigning a value to) each point of the plane by the highest value $h$ such that it appears within the boundaries of partition $P_h$. If we interpret these partitions as segmentations, we have a nice way of assigning importance to the contours thus produced. The problem now is to obtain a family of such segmentations. The *Waterfall algorithm* presents an intuitive solution.

The Waterfall algorithm was proposed by Beucher [18] as a means to morphologically build a hierarchy of nested segmentations using swamping or morphological reconstruction operator described in this section. The Waterfall is a hierarchical approach that iteratively selects the least contrasted among all the basins of the watershed topographic image. By swamping these basins and thereby eliminating the regional minima, a simplified partition is obtained. At the end, a single region corresponding to the global minimum, that spans the whole image is obtained.

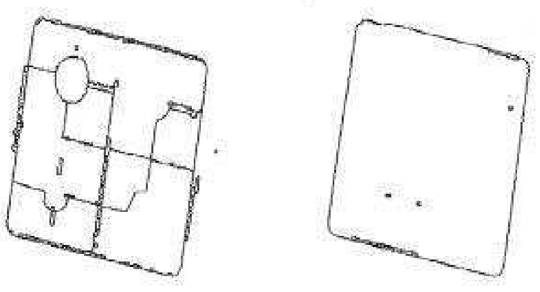
Let us invoke the very illustrative example that Najman and Schmitt [145] use to explain the change of dynamics along a hierarchical watershed. Every catchment basin with a regional minimum having a depth inferior to the level of the swamped dynamics has been eliminated until only the global minimum remains at 30 grey levels. The swamping levels have apparently been defined by the user and do not follow an automatic stopping criterion. Along with the increase in the dynamics level that is eliminated, the number of watershed region, i.e. over-segmentation from our perspective, steadily decreases.

The algorithm is non-parametric because instead of using markers provided by a previous process, it automatically selects significant regional minima by using



Nonconstrained watershed

Dynamics of 5



Dynamics of 10

Dynamics of 30

**Figure 4.5:** *Explicative example of the hierarchic watershed at various swamping levels from Najman and Schmitt [145].*

geodesic reconstruction by erosion (Annex $B$). We build a new topographic function $g$ by setting $g(x) = f(x)$ if $x$ belongs to the watershed, and $g(x) = +\infty$ if it does not. This function $g$ is obviously greater than $f$. Let us now reconstruct $f$ over $g$. It is easy to see that the minima of the resulting image are significant regional minima of the original image.

**Closing remarks** Some remarks are appropriate at this point. First, even if this procedure allows the convergence of a hierarchy by repeating itself until convergence, it does not allow a valuation of the watersheds thus obtained: the convergence is usually very rapid and only a few levels of hierarchy are present in the result. Second, even if we valuate the hierarchy by noting the order of extinction of the watershed, the final valuation result is only a manner of providing this partial ordering relationship and runs the risk of being independent of the underlying gradient information.

However, in the framework of our segmentation application, *the use of a partial labeling by fuzzy classification as the topographical function in lieu of purely gradient information makes the ordering relationship meaningful.* Since the vast majority cell or object basins are naturally deeper than all background basins, cell regions do not usually merge into the background.

The next section presents the manner in which we propose to remove the problems associated with the hierarchical watershed as well as of the marker-guided watershed within our segmentation framework, by synthesizing the two into an efficient and, as we prove in the later sections, a more accurate segmentation strategy.

## 4.3 Cumulative hierarchy

After having reviewed both the methods that prevent over-segmentation and improve segmentation quality or correct over-segmentation and improve over-segmentation over the baseline watershed segmentation, our quest for the most accurate segmentation strategy for extracting cells from our images has led to the conclusion that corrective methods are less effective in this case because of their use of innate image information, such as the weights on watershed lines, beyond our control and far from being optimally applicable. A good solution is generally inherently simple (Occam's razor [149, 67]) and elegant (consensus of the computer science community [102, 33]), i.e. it is the smallest and the most straightforward possible algorithm for producing the output that it does. We therefore re-visit the preventive strategies because of their simplicity and elegance and because they can integrate into the watershed algorithm instead of being an extraneous package. Reflection on the watershed transform and the immersion process,as described in the following, produced an elegant, more accurate as we see later, and less computationally intensive process, which we refer to as the *cumulative hierarchy*.

The watershed transform is a global transform, that is, it affects all parts of the image at once. A problem becomes apparent, because of this global nature, in the Waterfall or a successive-swamping algorithm. Parts of the image cannot be selectively swamped, and at any level $h$ of the hierarchy, all basins inferior to $h$ are swamped indiscriminate to whether they belong to what is known to be an object of interest, resulting in a loss of object regions shallower than $h+1$. Since the nature of this morphological operation cannot be changed, the overall segmentation algorithm should be so designed that these basins are preserved. Our goal is to approximate a way to localize the swamping, meaning that the hierarchy stops at different levels along different branches, as shown by the dendogram in Fig. 4.6. Thus, contrary to a global hierarchy that could produce for us a desired number of final basins, we could decide on the number as well as the choice of the basins.

Our solution takes advantage of the property 4.2.2.A and the particular topographic function while building a *bottom-up hierarchy*. Let us examine these factors individually in order to understand the mechanism of the algorithm. The waterfall definition implies that any basin that is swamped at level $h$ becomes part of a larger

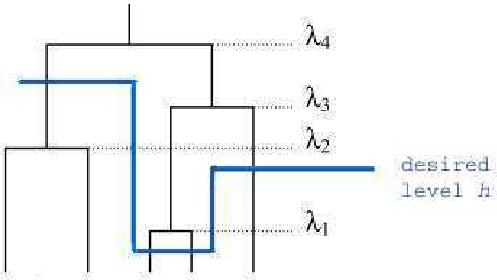


**Figure 4.6:** *Dendrogram representing a hierarchicy cut at not a single* $\lambda i$ *but at various ones according to application-mandated resolutions.*

one at level $h+1$. This implies that a region at level $h$ either merges into a larger cell region or disappears into the background at level $h+1$. The probability map that serves as the topographic function is such, as we have seen earlier in Section 3.5.4, that the topography of the cell is more extreme than that of the background and demonstrates deeper basins and higher crests. Thus non-cell basins will merge into the very large regions that the background produces at the initial levels i.e. for small values of $h$. Thus very early in the hierarchy creation of these background regions is complete whereas the cells are still fragmented into multiple regions. These regions within cells thus mostly absorb each other to grow into individual cells, while some at the fringes of larger ones are lost to the background regions. Ultimately individual cells form distinct regions, which persist through differing number of resolutions in the hierarchy until being lost to the background.

A local swamping could preclude the disappearance of interesting basins, i.e. the hierarchy would stop for a given basin and continue for other parts of the image. In the stead of preventing further resolution of the hierarchy we chose to “remember” any interesting region that does not exist beyond level $h$. Obviously the basins of interest are the ones containing a cell centre detected and validated from Chapter 2. Let us call all such watershed regions *centre-regions*.

All centre-regions can be identified, and extracted, from the watershed image at any level of the hierarchy. So they are identified at the $0th$ level and tracked along the upward progression of the hierarchy. The decimation of a centre-region is the result of its absorption either into another centre-region or into the background. This triggers region memorization. For the centre-region decimating into the background at level $h$ its state at level $h-1$ is copied into an accumulator image. For the centre-regions decimating into each other at level $h$ all of them at level $h-1$ are copied into the accumulator image. As the hierarchy is traversed from the bottom up, more and more centre-regions accumulate until finally the successively swamped watershed image is left with a unique background region. Through this process of *cumulative hierarchy* an image of distinct segmented cells is eventually arrived at. In the accumulator image, a centre-region at level $h+1$ replaces the one associated with the same centre but at level $h$, and by construction, all lower levels of the hierarchy, until its eventual absorption and disappearance.

Thus if at any level $h$, $f_h$ the topographic function for the watershed reconstructed by erosion over an image function which is its version shifted up by a constant step $h$, $C$ is the set detected cell centres, $BC_h$ the image resulting from the binary reconstruction of the watershed by $C$ at level $h$ and therefore containing only the set of watersheds that are absorbed at this level, and $Z_h$ is the set of topographic image pixels at level $h$, then the final topographic function is the the geodesic reconstruction by erosion $E^{\infty}_{f_h \bigwedge f}(f)$, and the watershed image at level $h$ will be:

- $f = P_{probmap}$
- For $h = 0$ to $N-1$
    - $f + h$ is reconstructed by erosion over $f$:
      $f = \psi^E_f(f+h))$
    - The watershed transform $W$ is calculated
    - Binary reconstruction by dilation of the watershed $W$ by the image of cell centres $C$ : $BC_h = \psi^D_C(W)$
    - When a basin $\in W$ disappears at level $h\prime$, its state at level $h\prime - 1$ is saved in an accumulator:
      $W_{accumulator} = W_{accumulator} \cup (BC_{h-1} \setminus BC_h)$

This algorithm allows us to obtain not only the exact number of regions as dictated by the number of counted cells but also their placement within the image.

Let us present some visual results that will lead to an appreciation of the working of the algorithm. Fig. 4.7 gives an overall view of the evolution of the cumulative hierarchy accumulator along the climb up the hierarchy. This is calculated over an entire image (image 0032), and the state of the watershed image at a given level of hierarchy is shown in the left column while that of the accumulator in the right column.

Fig. 4.8 shows the same algorithmic walk-through for a manually-selected rectangle taken from image 0032 so that small regions are easily viewable. Total decimation

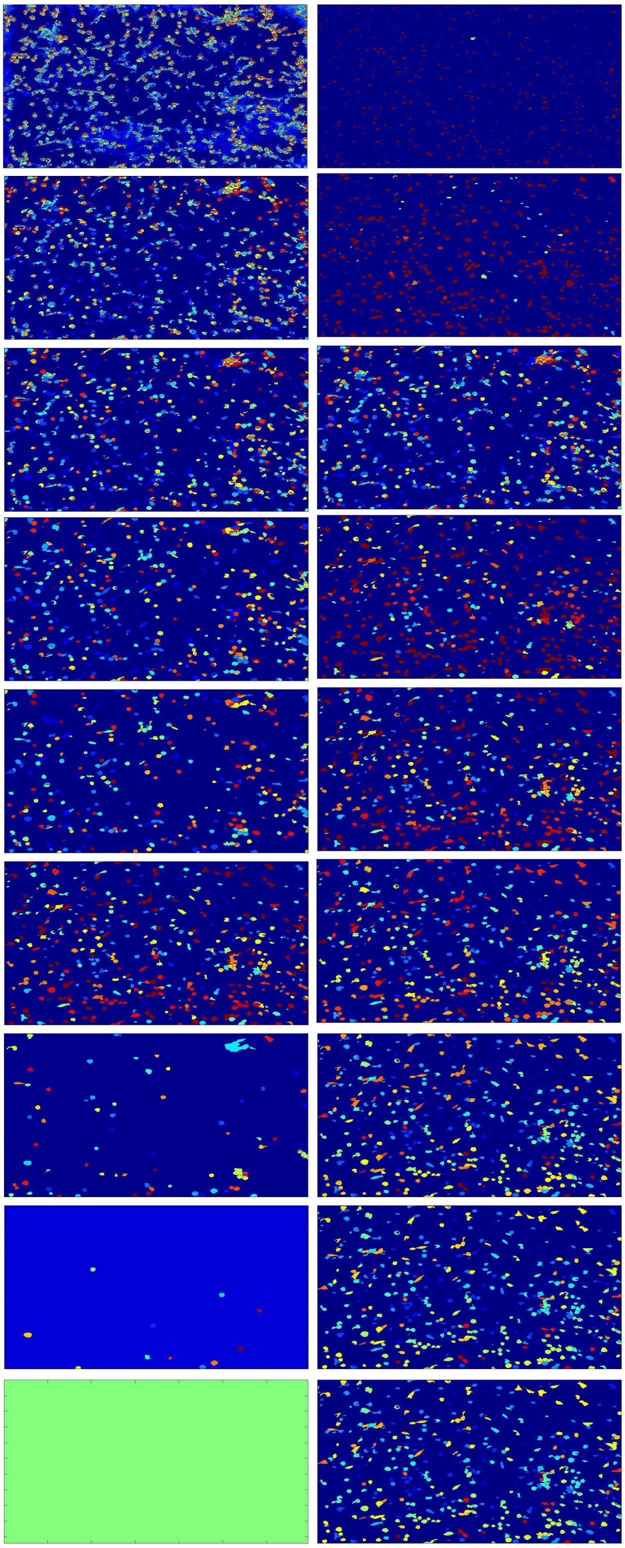
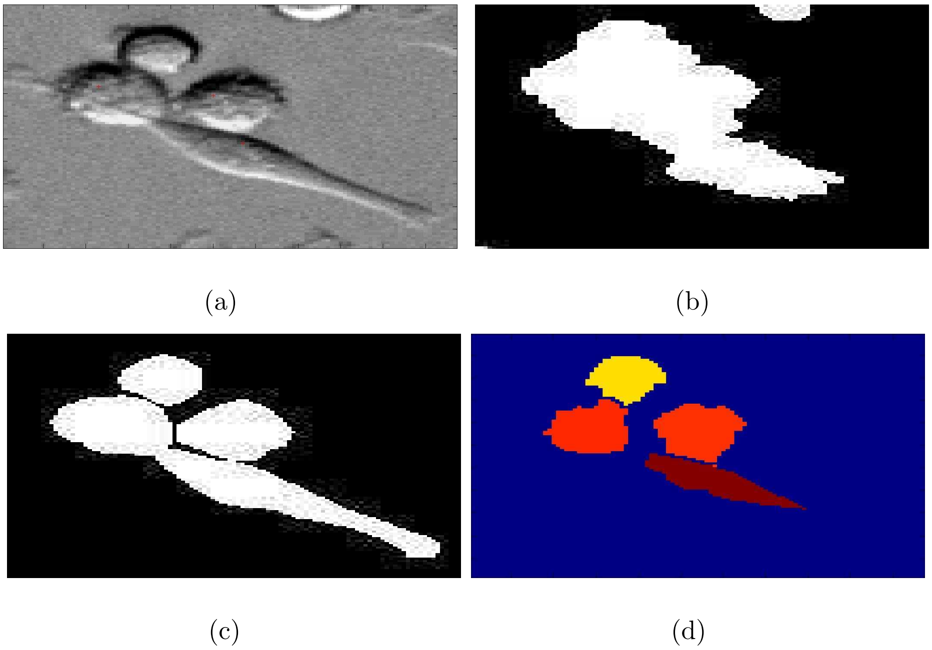


**Figure 4.7:** *Progression of the cumulative hierarchy. The column on the left describes the bottom-up traversal of the hierarchy and the one on the right the corresponding cumulation (levels 2, 22, 42, 62, 82, 102, 122, 142, 162).*

occurs at level 138, and every tenth level up to it from level 0 is shown. Again, the watershed images is on the left and corresponding accumulator ones on their right.

The following two figures place the reader in the process when viewed as a black box, i.e. inputs into and outputs out of it are described. Fig. 4.9 describes these inputs (4.9. a) and outputs (4.9. b) and offers the manually binarised imaged used in Segmentation Quality Evaluation as a benchmark in 4.9. c. A limitation of the cumulative hierarchy algorithm is also demonstrated in that it limits cell-regions to inside the Aggregate Connected Component in 4.9. c. in order to avoid Basin Overflow and hence loses potential regions that would have constituted the tail of that spread cell. Finally Fig. 4.10 offers the reader a comparison between cumulative hierarchy segmentation and the gradient weighted distance transform topographic image based watershed segmentation to expose a link to the previous work described in Section 3.5.2, because that was shown to be the next-best method in that set of conditions. The imprecision of the latter method is evident particularly in that the regions representing cells do not respect cell boundaries and eat into neighbouring cells and into the background, which would make discrimination between these difficult in the classification step.


(a) (b)

(c) (d)

**Figure 4.9:** *Details make interesting perspective: a) original subimage selection, b) the corresponding binarisation, c) the manual binarisation used for SQE, and d) the cumulative hierarchy segmentation produced.*

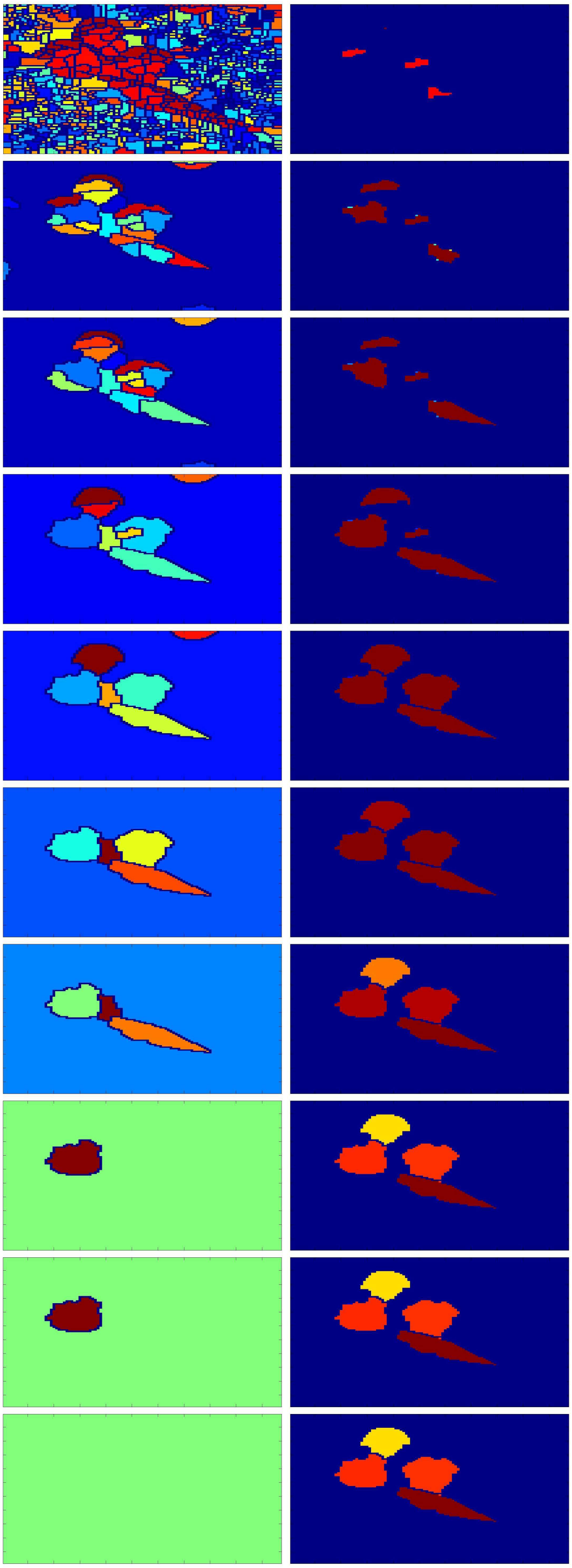
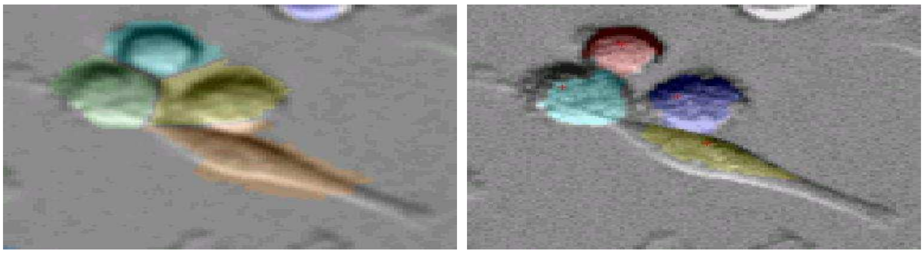


**Figure 4.8:** *Progression of the cumulative hierarchy. The column on the left describes the ~~bottom-up traversal of the hierarchy and the one on the right the corresponding cumulation~~ (levels 0, 8, 28, 38, 58, 68, 88, 108, 128, 138).*



(a) (b)

**Figure 4.10:** *Comparison of a) shape-guided gradient-weighted distance transform watershed and b) cumulative hierarchy.*

The bottom-up implementation proceeds thus: we modify the topographic image function at each iteration through *global geodesic reconstruction by erosion over itself shifted up by a constant level h* and this modified image function is segmented using a *watershed transform*. This process allows us the latitude of not defining a strict background marker that would not permit those basins that lie in its *influence zone* but potentially belong to cells. This, coupled with the characteristic of the probability-based topographic function that well-discriminates cell and background pixels, producing desired image regions corresponding to the extents of the centre-basins at each level of the hierarchy. At the same time, an *accumulator* image permits us to preserve the regions marked at inferior levels of the hierarchy but are absorbed into the background as the hierarchy tends toward the global minimum of the image, hence the name of the method. This method strictly solely uses *mathematical morphology*, and due to this quality, integrates seamlessly and elegantly into a framework that uses only morphological geodesic image reconstruction first by erosion and then by dilation and a watershed transform between the two. The reason for the selection of this method, however, is that it was able to translates this elegance of intuition into practical segmentation quality.

The usual methods of watershed segmentation improvement through post-processing take the form of *region merging*, which we introduce and discuss in the following section. It is there that the true usefulness of our cumulative hierarchy algorithm only becomes evident. In order to present a holistic appraisal of the situation between the algorithms we again broach further theoretic aspects the cumulative hierarchy in the conclusion of the chapter.

## 4.4 Correcting over-segmentation: Region Merging

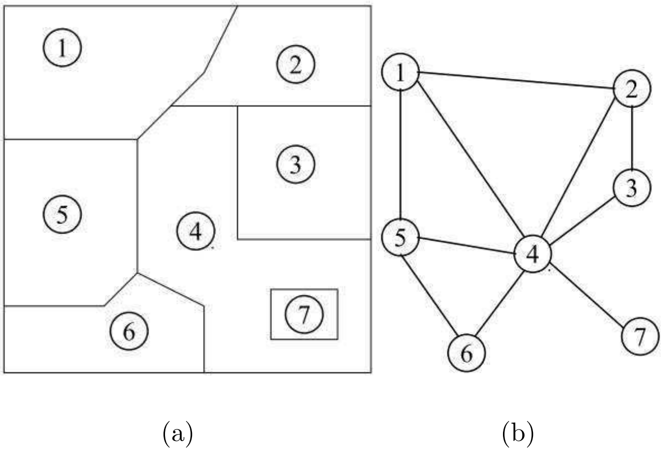
When watershed segmentation is applied to an image, water will rise from every minimum in the image, i.e., a unique label will be given to each image minimum. In many cases, not all image minima are relevant. Only the larger intensity variations mark relevant borders of objects. This means that applying watershed segmentation will lead to over-segmentation, i.e., objects in the image will be divided into several parts.

Over-segmentation can in theory be reduced by a pre-processing step reducing the number of local image minima, e.g., by smoothing the image with a mean or median filter, however these operations are not ensured to respect object geometries. Smoothing may remove important structures, such as edges, in the image. As is often the case, a post-processing alternative exists for this pre-processing. After applying watershed segmentation, over-segmented objects can be *merged*. *Merging*, or *region merging*, can be performed according to different rules, based on the segmentation model. One example is merging based on the height of the ridge separating two catchment basins, as compared to the depth of the catchment basins. The model says that a true separating ridge, must have a height greater than a given threshold. All pairs of lakes that at some point along their separating ridge have a height lower than the threshold are merged.

The following sections describe some usual methods employed in merging segmented regions or objects. The methods are discussed as presented in the corresponding literature and *implemented by us as possible routes to an improved image segmentation.* Wherever the implementation differs greatly from the original method presented because of a different application context or the fact that only the idea was adapted and a propriety implementation was devised, we also present the details of our algorithms. But first we will look at a representation of these regions that is used in several of these methods to implement the decision process in practice, the *region adjacency graph.*

#### 4.4.0.B Region Adjacency Graphs

A *Region Adjacency Graph* or *RAG* describes the *spatial adjacency* relationship between segments in a segmented image and is composed of a set of nodes representing connected components or regions of the image and a set of arcs connecting two neighbouring nodes. This *RAG* denoted by $G = (V, E)$ is constructed to describe a partition of the image by the topology and the inter-region relations of the image (Fig. 4.11). It is defined by an undirected graph where $V = 1, 2, ..., K$ is the set of nodes and $E \subset V \times V$ is the set of edges (denoting the presence of an adjecency relationship between two regions). $K = \theta(G)$ is the number of region nodes.



(a) (b)

**Figure 4.11:** *a) Example of the regions of a segmented image and b) the corresponding RAG.*

The *RAG* holds adjacency relations of regions which we need in order to merge those neighbouring regions that meet certain criteria as detailed in the following subsection.

#### 4.4.0.C Constructing the *RAG*

As mentioned before, a Region Adjacency Graph is the set $G = (V, E)$ that contains the connectivity or neighbourhood relationship among watershed regions or objetcs in general in a segmented image. Maintaining one is essential for many merging schemes, and could be done using data structures such as topological maps, but building a *RAG* efficiently could however be a tricky process. Several algorithms [183, 206] have been presented that outline the construction and management of the *RAG*, however one or more finer but crucial details always find themselves omitted. The principal such omission is the manner in which the adjacency relationship - which region is the neighbour of which region - is established. As has been proposed this can be done in a labeled connected component image by expanding outwards each labeled region until it encounters others, whose labels are noted, for example through morphological binary dilation; but this is an expensive operation, particularly when regions are abundant. Moreover, in our case the watersheds are labeled

```
Procedure: label segmentation lines and build RLRAG
Input:
segmented image of labeled connected components

Output:
RlRAG(region-region-line)
labeled line image

Algorithm:
Get list of coordinates of all line pixels

For each line pixel
    Find labels of touching connected components by running through
     the 8-neighbourhood,

    Remove line labels i.e. 0s,

      If neither connected components labels in RLRAG
       Label this line pixel,
       Put all 3 labels in RLRAG and the pixel's label
       on new line image
      Else
       Put line label for the region-region-line trio on the line image
       at this pixel,but don't update the RLRAG
      End If-else
End For
```

with the same value, 0, as the background, making the job even harder, particularly when we will also need the additional information of region-to-line spatial adjacency.

Therefore we have designed the following algorithm for constructing a *RAG* augmented with watershed-adjacency information, which is computationally efficient even in the event of a segmentation resulting in a very large number of regions i.e. in the tens of thousands, both in terms of time and memory requirements. The algorithm is independent of the segmentation method, and only requires a segmented image whose connected components have been labeled *a priori*. In addition to building a compact adjacency matrix, the procedure also labels the watersheds and returns this labeled image. The *RAG* is in the format $region - region - line$ i.e. in effect it is a **Region-Line-Region Adjacency Graph** *RLRAG*.

This is essentially a line-following procedure, where we traverse all pixels of the segmented image found on the separating lines. The rationale is that since a segmentation or a tiling of an image into connected components results in the vast majority of pixels being attributed to objects, processing the minority should result in a substantial speedup.

The other reason for the computational efficiency of this algorithm is the fact that for each image pixel, there is a lookup into the 8-neighborhood image matrix, which is much less computationally intensive than a morphological dilation operation. The image of the watershed lines thus obtained has been verified to have been created correctly i.e. with a different label for each line segment between any pair of connected components for the entire image set.

The *RLRAG* is used by any region merging method extraneous to and following the initial process of watershed segmentation, such as the ones we now present to the reader.

### 4.4.1 Criteria-based merging on the *RAG*

Section 4.4.2 will shortly discuss the criteria and the manners which are generally used for merging regions in an over-segmented watershed image. We use similar criteria and a global scheme to decide which regions merge and when does merging stop. Let us first look at the criteria.

***Global criteria***: Region merging for any given image is obviously also a function of the context in which the image was taken, other than being dictated by the aforementioned merging criteria. Indeed these context-derived criteria override the rest, since they take into account factors beyond the data extractable from the segmented image. For example, a region-merging scheme cannot be allowed to continue to merge regions until unity of regions when we know there is at least one object present in the image, even though traditional merging criteria so dictate. In our context, we possess knowledge prefatory to segmentation about the images: we have established a count of the number of cells in the image (Chapter 2), and thence determined their average size. This information serves well to be exploited at present.

Since we have found a certain number $\bar{C}$ of cell centres and have calculated the modal cellular radius and therefore the corresponding cell size (under the assumption of their circularity) $\bar{S}$ (refer to chapter 2 again), it is only natural that this knowledge be exploited if possible and if needed. Therefore, a global criterion is simply to ensure that the current number of image regions $C$ and the current average region size $S$ are as close as possible to the desired values mentioned above. That is we minimize the differences:

$$||C - \bar{C}|| + ||S - \bar{S}|| \tag{4.1}$$

***Local criteria***: The local criteria are taken from the discussion mentioned above. The ones we use are:

*Size*:

$$\frac{(\bar{S} - \text{area}_i)}{\max(\text{area})} \tag{4.2}$$

*Convexity*:

$$\frac{\text{conv}_i}{\max(\text{conv})} \tag{4.3}$$

*Depth*:

$$\frac{((\max(\text{depth}) - \text{depth}_i) \times \max(\text{depth}))}{\text{depth}_i} \tag{4.4}$$

where $\bar{C}$ and $\bar{S}$ represent approximated parameters actually calculated by the algorithm.

#### 4.4.1.A The initial algorithm and its shortcomings

As stated before, the procedure starts out on the basis of the current state of the watershed image with respect of the measures of two sets of criteria, and the information on the connectivity of the regions or watershed basins among them contained in the *RAG*. The merging procedure aims that the regions be as “cell-like” as possible in size, shape etc i.e. the local criteria, and therefore at each iteration merges the regions that violate the set of local criteria. The procedure goes on until the global criteria are met i.e. the number and the general size of the regions does not begin to diverge from the manually-labeled image. A region, once candidate to merging according to local criteria, merges with its neighbour with whom it has the weakest border according to Wälby’s definition i.e. having the smallest mean height. Once a new region is formed it is assigned a new region label. The *RAG* is updated at the end of each iteration to reflect the state of the connections between regions that were removed and the new ones formed with the new regions.

In this way vast swathes of the image are updated at each iteration, but it comes at the cost of accuracy. The very apparent problem with this approach is its use of an implicit set of thresholds, particularly in the case of the local criteria. These thresholds are inevitable because the criteria as defined above produce each a score in the interval $[0, 1]$. The thresholds are determined from the scores of the regions that constitute whole cells found outside of cell unclusters. But these supposed cell-regions might not themselves be perfect, e.g. formed of a cell divided into multiple regions, and risk producing, if only slightly, different criteria scores thanks to being not surrounded by touching cells. The local criteria are *AND*ed amongst themselves on their individual thresholds decisions. Therefore, to avoid as much as possible the summing of uncertainties induced by each threshold, we have used a few but significant criteria. Still, convergence could be elusive for the global criteria since between two iterations a large number of regions merge and fine-tuning the procedure globally is not possible. There is thus a theoretical schism between the global and the local criteria sets and the local ones allow merging of numerous regions in one iteration, causing the global criteria to over- or under-estimate their objectives by a significant margin (in tens of cells) in the final iteration.

A second problem is the erratic manner in which regions might merge. Any pair of regions merge based on which watershed line is the weakest between a region and its neighbours. While this is a natural and intuitive way of watershed line breaking, it does happen to merge regions across cell boundaries, a quality attribute established by the human eye. That is general problem that we will not attribute to this method, but it is a problem that we will try to minimize, as we see shortly.

Another discrepancy is the ordering affects in region merging due to the order of traversal of the *RAG*, but this too is inevitable. The *RAG* is accessed in the order of region discovery or labeling in the image i.e. from top left to bottom right. Once a region is found to be the candidate to merging as a result of its local criteria scores, it is merged with the connected neighbour connected by the weakest border or the watershed line having the smallest minimum value along it. The new region thus produced is assigned a new label, has its local criteria scores calculated, and is inserted into the *RAG* in place of the two former regions that now constitute it. But a different order of traversal of the *RAG* may not even consider this pair of regions for merging because one or both of them might have been merged with neighbouring regions encountered earlier in *RAG* traversal. Thus there is no global ordering of regions, and flipping the image around or using a different labeling order will change the merging result. This also means that the newly-created regions are studied for possible merging towards the end, owing to their higher region labels, which is a desirable property since new regions are assumed to be more “cell-like”.

#### 4.4.1.B Our improved basin-line competition implementation

In the algorithm presented above we saw several drawbacks:

- setting threshold on the local criteria, which complicate and often hinder correct merging,
- rapid and abrupt region merging often with lack of global convergence,
- the selection of the pair of regions to merge in a connected neighbourhood,

- the order of merging.

We decided to improve the algorithm mentioned above to eliminate these flaws. Hence the method we discuss in this section is a derivative of the previous one.

One possible manner to resolve the issue of local thresholds is, instead of comparing quantities describing an object to calculated thresholds, they be compared with those emanating from other objects. Thus a competition between several connected objects would decide the winners.

Watershed regions and lines present a conceptual duality: strong regions can exist without being captured by neighbours despite being surrounded by weak watershed lines, and strong watershed lines, purportedly representing object borders, should put a hold to the merging of two neighbouring regions that may be eligible to a merge according to any set of criteia. The notions of weakness and strength can be defined either as in section 4.4.4 or though Wählby's criteria [219] as described in Section 4.4.3. Thus we think of the region and the line in an antagonistic relationship, and propose the following: a watershed region has the propensity to merge with its neighbours, and a watershed line has an inertia to breaking i.e. a tendency to remain. The latter is a straightforward corollary of the line-strength definition. The former behaviour can be formulated with criteria that encourage regions that are not "cell-like" to merge.

There is only one global criterion, the number of cells, which the number of regions will finally try to match. A number of local criteria are defined, both for regions and for watershed lines.

**Local region criteria:**

*Size*:

$$\frac{(\bar{S} - \mathrm{area}_i)}{\bar{S}} \tag{4.5}$$

The smaller (removed from ideal cell size) is the cell region the higher the merging propensity.

*Convexity*:

$$\frac{\mathrm{conv}_i}{\max(\mathrm{conv})} \tag{4.6}$$

The less convex is the cell region, the higher is the merging propensity.

*Depth (lack of)*:

$$\frac{((\max(\mathrm{depth}) - \mathrm{depth}_i)}{\max(depth))} \tag{4.7}$$

The shallower the cell region is, the higher the merging propensity.

The three criteria are summed and divided by 3 so that the Net Merging Propensity of a region is bounded by $[0, 1]$.

**Local line criteria:**

*Mean Depth*:

$$\frac{((\max(mean(\text{depth})) - mean(\text{depth})_i)}{\max(mean(\text{depth}))} \tag{4.8}$$

The deeper the watershed line the higher the inertia.

*Variation along the line*:

$$std_i^{-1} \tag{4.9}$$

to go along with the mean value, the higher the inverse of the grey level standard deviation along a line the higher its inertia.

*Variation along the line*:

$$\frac{(2 \times \pi \times \text{ modal cell radius} - \text{length}_i)}{\max((2 \times \pi \times \text{ modal cell radius} - \text{length}_i))} \tag{4.10}$$

The closer to a semicircle the region bounded by the watershed line is, the higher the inertia.

Thus for the segmentation to be optimal, we once again wish for the regions to be as “cell-like” as possible, as defined by the criteria. The algorithm proceeds as follows: the regions wish to attain stability by being more “cell-like” through merging, and the lines wish to remain intact to not allow the regions they enclose if the latter are already “cell-like”. Thus the sum of the two sets of criteria are normalized to 1 for each region and each line, and the $RAG$’s edges are weighted with the difference $WL_i - \text{inertia} - 0.5 \times (R_j - \text{merging-propensity} + R_k - \text{merging-propensity})$. At each iteration, the edge representing the lowest weight over the entire $RAG$ is broken, the regions merged, the newly-created region assigned a new label, its merging propensity calculated and its related edges updated.

The problem of convergence is resolved by only allowing one merge every iteration of the algorithm. Because we now have a global view of the tendencies of region-pairs to merge through their candidacy-scores, we select the one with the highest score for merger. In this manner, region-merging happens in a manner that does not negate the goals we set for the image globally. But this also means sacrificing processing efficiency for merging accuracy: a higher number of iterations is required, each necessitating $RAG$ traversal and update.

The problem of deciding for a region with which neighbour to merge is automatically solved because this choice is now made globally, and because this choice is now the result of a comparison between quantities which are more computationally stable than a simple minimum over a watershed line.

The issue of the global ordering of regions is also resolved by the ordering relationship determined by the weights on *RLRAG* edges.

Once again, this is a computationally-expensive operation, and we must be careful not to apply it over a watershed image but when the over-segmentation is not very acute.

### 4.4.2 Model-based Object Merging methods

To overcome the problem mentioned in the introduction of this section, some type of merging mechanism has to be introduced in the post-processing step. Several techniques have been proposed in the literature. One possible method is to make use of hysteresis thresholding to filter noisy weak contours, representing the watershed lines between small regions. As pointed out by [145], hysteresis thresholding produces non-closed contours and barbs in the case of watershed. [3] presented a rule-based heuristic merging technique to reduce over-segmentation, by identifying the oversegmented objects based on size, and merging them with their parent nucleus. This method represents a significant advance, but can be improved upon. Its limitations arise from the fact that merging purely based on object size is prone to error, especially when segmenting objects with great variation in size. Second, a global size threshold is not easy to set in an automated and consistent manner. Finally, the merging rule does not account for the features of other objects in the image.

Statistical shape-modeling methods depend upon the availability of parametric models to describe the nucleus objects. These parameters must be selected carefully in order to accurately characterize the nucleus objects, and discriminate outliers from real nucleus objects in an effective manner. The set of parameters must be rich enough to describe complex objects. A realistic strategy for estimating these parameters is for the user to specify examples of valid and invalid nucleus objects, and to perform supervised morphometry on these objects. In practice, the tedium and labor cost of specifying these examples is high enough to limit the number of examples. This, in turn, forces us to limit the number of object modeling parameters.

Let the location of the pixels in a cell nucleus denoted $p = \{p_0, p_1, \ldots, p_{n-1}\}$, where $p_i = \{x_i, y_i, [z_i]\}$. Their corresponding pixel intensity values are denoted $v = \{v_0, v_1, \ldots, v_{n-1}\}$. The following features could be readily measured.
*Volume:* The volume (size) of the object, $V$, is the total number of voxels inside the

object, i.e., $V = n$.

*Texture:* The simplest texture measure, denoted $T$, is the standard deviation of intensities of all pixels inside the object

$$T = \sqrt{\frac{1}{n-1}\Sigma_{i=0}^{n-1}(v_i - \bar{v})^2} \tag{4.11}$$

where $\bar{v}$ denotes the average nucleus intensity.
*Convexity:* The convexity, $S$, of an object is defined as the ratio of the object volume to the volume of the convex hull of the object. The convex hull of an object can be formed by a method called *Jarvis's March* [92]. The convexity is desired to be close to one for circular and elliptical objects, and less than one for concave objects.

*Shape:* Let $Q$ be the boundary pixels of the object. The shape feature, $U$, is defined as

$$U = \frac{|Q|^3}{64\pi \times V^2} \tag{4.12}$$

where $|.|$ denotes the number of elements in a set.

*Circularity:* Let $\bar{p}'$ denote the purported center of the object, then the distance between pixels $p$ and the center can be described as $d = ||p' - \bar{p}'||$. The circularity, $C$, is defined as

$$C = mean(d)/stddev(d). \tag{4.13}$$

*Area:* The area, $A$, is the number of pixels of 2D objects, i.e., $A = k$.

*Mean radius:* Let $R$ be the vector of the distances from the boundary pixels to the center $\bar{p}'$, and the mean radius $\bar{R}$ is defined as the average of $R$, i.e., $\bar{R}$.

*Eccentricity:* The eccentricity, $E$, is defined as the ratio of the major axis to the minor axis, and can be estimated by the ratio of the maximum to minimum radius $R$, i.e., $E = max(R)/min(R)$.

The statistical object model is an $m$-dimensional Gaussian distribution defined on a vector of $m$ features $X = (x_1, x_2, \ldots, x_m)$ drawn from the list above. The distribution requires the mean, denoted $\bar{X}$ and covariance matrix, denoted $\Sigma_X$ . These parameters are estimated from a subset $C_t$ , of the objects $C$ produced by the watershed algorithm.

The training set $C_t$ is selected as follows. It is known that objects representing intact cells in these results are generally characterized by a relatively large value of volume $V$, convexity $S$, and circularity $C$. Based on these considerations, the training set can be constructed by placing thresholds on volume $V$, convexity $S$, and circularity $C$, as described below:

$$C_t = \left\{ c | c \in C, V_c \geq overlineV + t\sigma_V; S_c \geq \overline{S} + t\sigma_S; C_c \geq \overline{C} + t\sigma_C \right\}, \tag{4.14}$$

where $\bar{V}$, $\bar{S}$, and $\bar{C}$ are the mean values of object volume, convexity and circularity, and $\sigma_V$ , $\sigma_S$ and $\sigma_C$ are their corresponding standard deviations respectively, $t$ is an empirically specified parameter that sets the degree of selectivity.

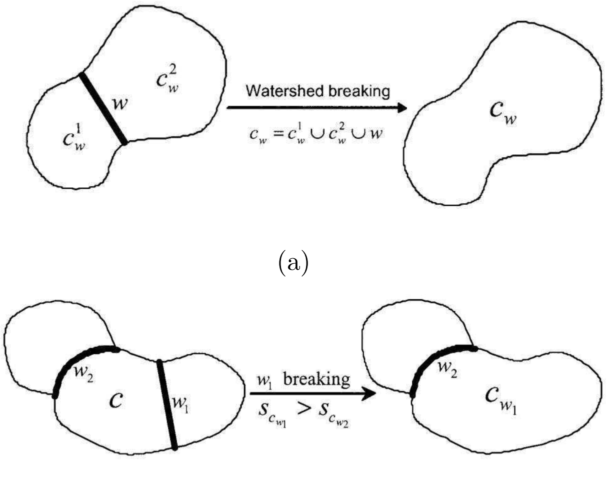
Based on the above Gaussian model, we can measure the confidence score for any given object $c$ with feature $X$, using the Gaussian probability that the object feature fits the model, as follows:

$$S_c = p(X) = \frac{1}{(2\pi)^{m/2}|\Sigma_X|^{1/2}} exp(-\frac{1}{2}\sqrt{(X-\bar{X})^T \Sigma_X^{-1} (X-\bar{X})}). \tag{4.15}$$

To correct the over-segmentation produced by the initial watershed over the probability map, it is necessary to detect and eliminate the false watersheds and thereby merge segmented objects. This is guided by a merging criterion based on a merging score derived from the confidence measure described above in equation 4.15. Let $W$ denote the set of watersheds that separate adjacent objects, in our case cells. As illustrated in Fig. 4.12 a., each watershed surface $w \in W$ separates two touching nuclei, denoted as $c_w^1$ and $c_w^2$. We define the gradient of $w$ as the average intensity gradient among all pixels on the watershed $w$, i.e., $\lambda_w = (\Sigma i \in w\lambda_i)/n$, where $n$ is the number of pixels in $w$. In the same manner, we define the intensity gradient $\lambda_{cw}$ for each object $cw$ by averaging the intensity gradients among all pixels in $c$. Let $c_w$ denote an object formed by breaking $w$ (in other words, merging $c_w^1$ and $c_w^2$ separated by w). Then, we have:

$$c_w = c_w^1 \cup c_w^2 \cup w \tag{4.16}$$

Note that pixels corresponding to the watershed surface $w$ itself should also be merged into $c_w$. The confidence score of $c_w$, based on equation 4.15 above, is called the "merging score" and denoted $S_{c_w}$ in the following. Intuitively, the merging decisions are based on the following two observations: 1) The merging score $S_{c_w}$ should be higher than the score of either nucleus before merging, i.e., $S_{c_w}^1$ and $S_{c_w}^2$. 2) The gradient of $w$ should be relatively large compared with the gradient of objects or regions $c_w^1$ and $c_w^2$. This is based on the assumption that intra-region gradients



(a)

(b)

**Figure 4.12:** *Illustrating two example cases encountered by the watershed-breaking algorithm. (a): A case that leads to merging of the two objects. (b): The case where one object has multiple (two) watershed lines. In this case, there are two candidate watershed surfaces to choose from for breaking. Our algorithm prioritizes the watershed surface* $w$ *that has a greater merging score* $c_w$*, indicating better fit to the object model, thus the higher confidence towards its breaking. Explicative figure courtesy G. Lin [119].*

are smaller than inter-region gradients, which generally holds true. With these observations in mind, we calculate the following ratios:

$$R_{S_w} = \frac{2 \times S}{(S_{c_w}^1 + S_{c_w}^2)} \tag{4.17}$$

$$R_{\lambda_w} = \frac{(\lambda_{c_w}^1 + \lambda_{c_w}^2)}{2 \times \lambda} \tag{4.18}$$

The ratio $R_{S_w}$ reflects the relative degree that the objects match the statistical model before and after merging, thus it accounts for the confidence we have on the breaking of $w$. The higher $R_{S_w}$ is, the more confidence we have in merging $c_w^1$ and $c_w^2$. The ratio $R_{\lambda_w}$ captures the intuition that a watershed with high intensity gradient is likely the boundary of two touching nuclei. The higher the $R_{\lambda_w}$, the less likely that $w$ represents background pixels, thus more likely that $w$ belongs to the interior of an object, rather than $c_w^1$ and $c_w^2$ being two objects separated by $w$. The above two ratios can be combined as follows into a single decision making criterion:

$$R_w = R_{S_w} \times R_{\lambda_w} \geq \beta \tag{4.19}$$

where $\beta$ is an empirical decision threshold (typical value *1.2* [119]).

Breaking of the watershed $w$ results in the merging of two objects $c_w^1$ and $c_w^1$. This procedure is repeated until no more watershed surfaces in $W$ satisfy the condition in equation 4.19. Special attention needs to be given to nuclei that touch more than one object, as illustrated in Fig. 4.12 b. In this case, we have multiple candidate watershed surfaces to be selected for breaking. Intuitively, we must assign a higher priority to the one that has a greater merging score, i.e., break the watershed with the greatest $c_w$ value before other watershed surfaces.

### 4.4.3 Watershed-line breaking methods

Watershed segmentation is expected to result in over-segmentation. The segmentation is therefore followed by two merging steps. The first merging step removes extra regions due to non-seeded local minima. These regions are merged with the neighboring region towards which it has its weakest border. The weakest border is defined as the border in which the mean intensity of the inverse of the original image is the least [219]. The seeded neighbor may be either object or background, and the merging continues until all non-seeded objects are merged. The second merging step deals with over-segmentation resulting from extensive $h$-maxima seeding. This over-segmentation is reduced by removing region boundaries crossing bright parts of the image, e.g., a boundary dividing a bright cell in two. In this case we continue the merging until all remaining objects have a defined maximum average intensity along the border (greater or smaller than that of the basins depending on the topographical image). This step will not only reduce over-segmentation, but also merge false objects, such as debris, with the background.

After watershed segmentation, over-segmentation was reduced by only keeping those borders that correspond to strong edges. If two seeds are in the same object, the magnitude of the gradient at the region border will usually be small.

Associating region boundaries with border strength requires some careful definitions. The strength of a border separating two regions should be calculated in such a way that the strength of the border between regions $A$ and $B$ is the same as the strength of the border between $B$ and $A$. This is achieved by the following method, where the image of the segmentation result is traversed once. If the current pixel/voxel has a label which is different from that of a "forward" neighbor, (2 edge and 2 vertex neighbors in the $2D$ case), the pixel intensities from the corresponding two positions in the gradient magnitude image are retrieved. The intensity of the brighter of the two is used to define the local border strength between the two neighboring pixels and saved in a table for border data. We choose the brightest value since it represents the strongest border value. If a pixel has several forward neighbors with different labels, each label will result in a new value in the table of border data.

The strength of the complete border between two regions can be measured in many different ways. A simple measure is to define the strength of a border as the weakest point along the border. This is often used for reducing over-segmentation resulting from watershed segmentation. However, many correctly segmented objects are then merged, due to single weak border pixels or weak border parts originating from locally less steep gradients. Another measure, which is less sensitive to noise and local variations, is the mean value of all border pixels of two neighboring objects. This measure proved to give good results for merging of over-segmented nuclei. The mean value of the border of each merged object must be updated after merging. This is done by adding the border data (mean value and number of pixels) of the merged objects to the new, larger, object and its neighbors. The merging is continued until each remaining object border is stronger than a given threshold. Instead of defining the border strength as a mean value, one might consider the median, or some other percentile, but this would need storage of more data than just the number of pixels and the pixel sum.

A strong border means that the object is well-focused. Merging based on border strength therefore means that not only over-segmented objects are merged, but also poorly focused objects will be merged with the background, and disappear. This is an important feature if well-focused objects are required in the further analysis of fluorescent signals. In this case, a rather high threshold is suitable. If also poorly focused objects are of interest, their removal can be avoided by not allowing merging of objects and background.

This method is not only intuitive but also straightforward to implement, albeit prone to noise-induced-variation in the watershed-line score. However, in our implementation we use the mean value instead of the minimum, affording more stability if less correctness (basins spill once they are flooded above the minimum value on any encircling watershed line) to the method. Moreover, the method has a similitude with the hierarchic segmentation, only it is the watershed lines instead of the basins that are successively eliminated.

### 4.4.4 Significance-of-basins approaches

In their seminal work, Léon and Bleau [25] address the following question: given an image $I$ , what is the transformed image $T[I]$ that is closest to $I$ but has fewer and more significant watersheds? Obviously, the answer will depend on the exact meaning we attach to “closest” and “significant”. In the following, we discuss the use of several specific criteria to substantiate these words.

**Significance criteria**. To assess the significance of watersheds, the simplified algorithm will rely on some measurements. The selected criteria may be:

1. Watersheds with depth reaching a threshold $D_t$ are considered significant. Threshold selection has to take into account such factors as grey-level digitization method and enhancement pre-processing performed.

2. Watersheds with core surface reaching a threshold $S_t$ are considered significant. Threshold selection has to take into account such factors as scaling, and zooming.

3. Watersheds with volume reaching a threshold $V_t$ are considered significant. Threshold selection has to take into account the factors of both measurements described above.

4. A combination obtained by OR-ing or AND-ing the above criteria, such as
   $Surface[WScore[I, WS]] \geq S_t$ AND
   $Depth[I, WS] \geq D_t$ $Volume[I, WS] \geq V_t$ OR
   $Depth[I, WS] \geq D_t$ $Volume[I, WS] \geq V_{t_1}$ AND
   $Depth[I, WS] \geq D_{t_1}$ OR $Volume[I, WS] \geq V_{t_2}$ AND
   $Depth[I, WS] \geq D_{t_2}$,
   with $V_{t_2} > V_{t_1}$ and $D_{t_2} < D_{t_1}$. The criteria listed above were selected for the simplified algorithm because all of them are dependent on the overflow altitude of the watershed under study.

**Additional significance criteria**. Several other criteria may be used to assess the significance of watersheds; some involve a priori knowledge in the form of markers, and some involve measurements.

1. Markers-based criteria: Let $Markers$ be a set of pixels known *a priori* to be located on significant features. This set may be the union of several disjointed connected components, $Markers\ k$, that we will term marker components.

   (a) Watersheds touching a marker are considered significant, i.e., the watersheds $WS$ such that $WS \cap Markers \neq \emptyset$ .

   (b) Watersheds including a marker component are considered significant, i.e., the watersheds $WS$ such that $\forall k |\ Markers_k \subseteq WS$.

2. Measurement-based criteria: Watersheds with total surface exceeding a threshold $St$ are considered significant. Threshold selection has to take into account such factors as scaling and zooming.

These criteria may be combined with first set of criteria in various ways. If the selected criterion includes terms that are independent of the former, such as a minimum surface or the inclusion of markers, some of the previous assumptions about which watersheds will never satisfy that criterion will not hold.

Frucci et al. [64] [65] simplifies the significance-of-basin approach by studying the significance of basins in relation with their neighbours. They consider the significance of a basin $X$ as depending on the interaction of $X$ with its adjacent basins. In this way, we are able to select the basins with which merging of $X$ is more convenient. Such an interaction is evaluated by taking into account some features of the basin $X$ and of its adjacent basin. First, let us define the relative significance of $X$. We say that a basin X is significant with respect to an adjacent basin $Y$ if the following holds.

$$SA_{XY} > A_t \qquad or \qquad D_{XY} > Dt, \tag{4.20}$$

where the thresholding values $At$ and $Dt$ are computed by taking into account the initial watershed transform and by analysing the frequencies of the similarity parameters and local depths associated to the basins. We will also say that the $WL$ dividing $X$ from $Y$ is a strong separation with respect to X when (4.20) holds. Then, we define the intrinsic significance of $X$ in terms of the relative significances of $X$, and distinguish three degrees of significance:

*Strong significance:* if any $WL$s surrounding $X$ is a strong separation with respect to $X$,
*Weak significance:* if no $WL$ surrounding $X$ is a strong separation with respect to $X$.
*Partial significance:* in all other cases.

Frucci et al. propose a merging iterative process in which each iteration consists of two steps, respectively applied on weakly significant basins, and then on partially significant basins. In the first step, they assume that a weakly significant basin $X$ can be merged with any of its neighbours, and to this purpose we use a flooding transform. Since some basins, resulting from a single flooding process, may still be weakly significant, this step is iterated until only partially and strongly significant basins are present in the image. In the second step, the removal of a partially significant basin $X$ is accomplished by merging $X$ with its neighbours belonging to

$NS(X)$, where $NS(X)$ is the set of adjacent basins which are non significant with respect to $X$ (and with respect to which $X$ is non significant) and whose regional minimum are at altitude not greater than $PR_X$, the corresponding regional minimum. If $NS(X)$ is not empty, a steepest descending path is created starting from $R_X$ and terminating on $R_Y$ , for any basin $Y$ of the current $NS(X)$, and the watershed transformation is repeated. Due to the modifications of the degree of intrinsic significance of a basin occurring during the removal process, the whole merging process needs to be iterated to obtain basins which are all strongly significant. Note that the initial values computed for $At$ and $Dt$ might be no longer valid for the final image, and have to be computed again. If one of these new values ($At'$ or $Dt'$) results greater than the previous value then the merging process is applied again. The whole process terminates when neither $At$ nor $Dt$ are greater than the previous values.

## 4.5 Cumulative hierarchy versus the other segmentation-improvement methods: Evaluation and discussion

We wish, at this point, to explain the emphasis that we give to cumulative hiearchy with respect to the other methods, in particular to the basin-line competition algorithm which too has been developed by us. The cumulative hierarchy is a method that prevents over-segmentation rather than being a post-segmentation operation. Just as well, it modifies the image terrain at each iteration and at the same time control the further unification of the regions in the watershed image according to criteria similar to those presented in this section for region-merging methods. Therefore, by its qualities of integration into the watershed transform, of the intuitiveness and the elegance of the method itself, cumulative hierarchy is our method of choice at the theoretical level. In this sub-section we analyze whether the theoretical preference translates into practical segmentation quality. For this purpose, we return to the $SQE$ methodology presented in Section 3.3.

The comparison has been carried out in the same manner as earlier in section 3.5.5. The test data comprises 30 cell-aggregate connected component *subimages*. Each has been extracted from our test set of 14 images, and therefore represent well the variability in the image data. The aggregation within most of the connected components also means that the simpler case of an individual cell in a subimage has not been considered, but some connected components do have non-adhering cells, hence incorporating that case in the study.

The *ground-truth* (refer to section 3.3.2) is established by tracing out the contours of each cell in a subimage by hand. We start with a lasso that is unattached to any point in an image and anchor it to every judged salient point on the cell boundary. No interpolation is performed between any pair of these points and the lasso is non-rigid i.e. we end up with a piecewise-linear curve.

The merging algorithms being compared are from among those described earlier in this section. Specifically, they are:

- Waterfall with global stopping criteria on the *RLRAG* (Section 4.2.2.A)
- Region-line competition merging (criteria-based merging on *RLRAG*) (Section 4.4.1.B, improved algorithm)
- Object-Model Merging through watershed breaking over the *RLRAG* (Section 4.4.3)
- Merging on basin properties according to Léon and Bleau (Section 4.4.4)
- Cumulative Hierarchy (Section 4.3)

Let us briefly recapitulate these methods in Table 4.1 for ease of reference in the following discussion.

The last three methods are repeated without an initialization, thus we have a comparison among 8 different applications of 5 individual methods, the last three separated for better readability into Tables 4.5, 4.6 and 4.7 which offer a comparison for these merging methods wit and without the aforementioned initialization. The initialization we refer to is a certain level of hierarchy in a waterfall segmentation that is applied to reduce the number of regions to merge, and to ensure that too many regions are not erroneously merged across object boundaries. This level of hierarchy is constant for each algorithm, and is the cumulative hierarchy optimum for its global criteria. Each *subimage* is segmented at this level the first five times and at level 0 the last three, and then passed onto the corresponding merging algorithm. At the end, one region representing each cell is selected using a marker. As before the marker provides a *ceterus paribus* situation for comparison, but conduces to a very strict comparison since the regions left unmerged but still inside the aggregate connected component are not taken into account. This phenomenon is especially prominent in the without-initialization implementations because the algorithms stop much earlier than the optimum of cell-like regions, and we have very significant Basin Shortfalls.

| | | |
|---|---|---|
| 1 | Waterfall, global stopping criteria | Classical waterfall but with examination of the number and the sizes of cell regions at each level in order to stop. |
| 2 | Region-line competition | Merging on the *RAG* using local criteria for watershed regions and lines so that regions are mergeophile and lines mergeo-phobe, and for each edge of the *RAG* one wins over the other, until stability. |
| 3 | Watershed breaking | Defining statistical similarity between regions as *RAG* nodes and testing whether eliminating *RAG* edges increases similarity scores for merged regions over their former component regions. |
| 4 | Léon and Bleau | A significance-of-basins approach that weighs *RAG* edges similarly to 2 and decides which basins are strongly, partially and weakly significant, and iteratively filling up the latter ones and dissolving *RAG* edges until only strong significances remain. |
| 5 | Cumulative hierarchy | Marks the topographic function at increasing levels with validated cell centres, and accumulates associated basins as they extinguish. |

**Table 4.1:** *Recall and summary of the segmentation refinement methods discussed.*

Each of these marked cell-object are compared to the ground-truth according to the discrepancy criterion in 3.3.2. Figures 4.13 through illustrate the comparisons.

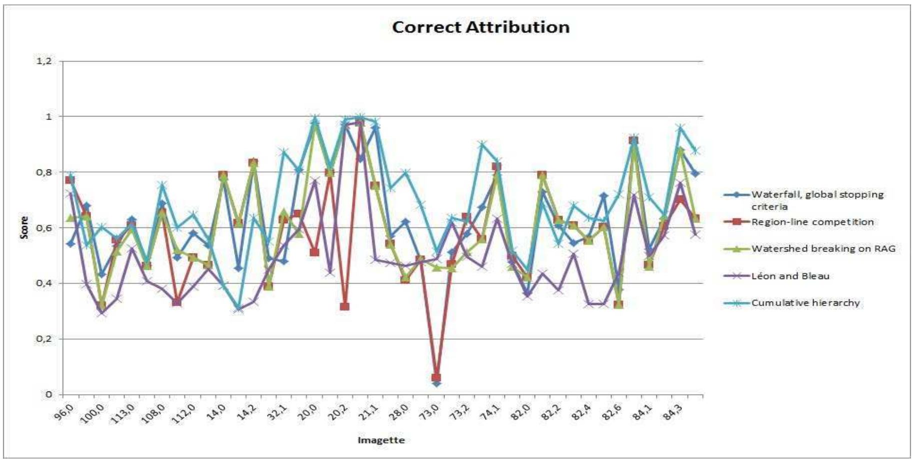
We present the set of figures 4.13, 4.14 first and then discuss the conclusions. Fig. 4.13 compares (a) correct segmentation (CA), (b) basin overflow (BO) and (c) basin shortfall (BS) for the *subimage* set.

**Table 4.2:** *Mean and standard deviation of Correct Attribution by improvement algorithm.*

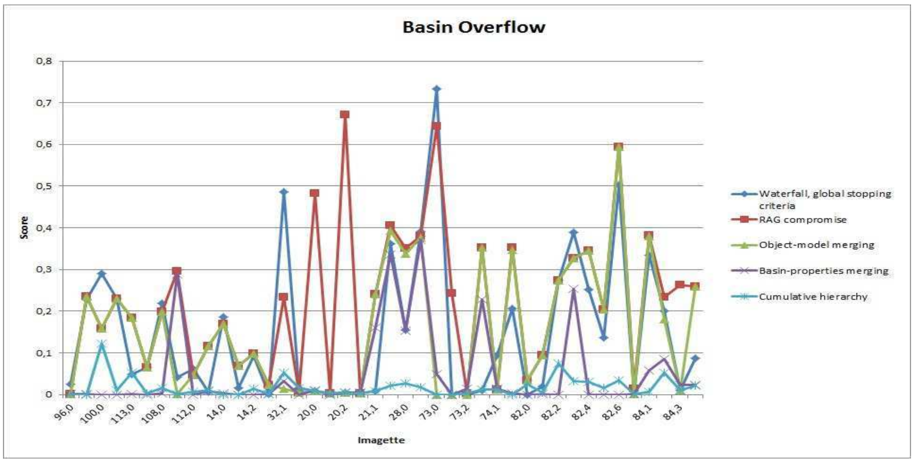
| Method | Waterfall, global stopping criteria | Region-line competition | Watershed breaking | Léon-Bleau | Cumulative hierarchy |
|---|---|---|---|---|---|
| Mean | 62.77% | 57.81% | 61.34% | 49.89% | 69.56% |
| Stdev | 19.06% | 18.14% | 17.50% | 16.25% | 17.27% |

- The comparison is self-evident in that the first three of the five algorithms outperform the rest by a significant margin, with respect to all three quality criteria.

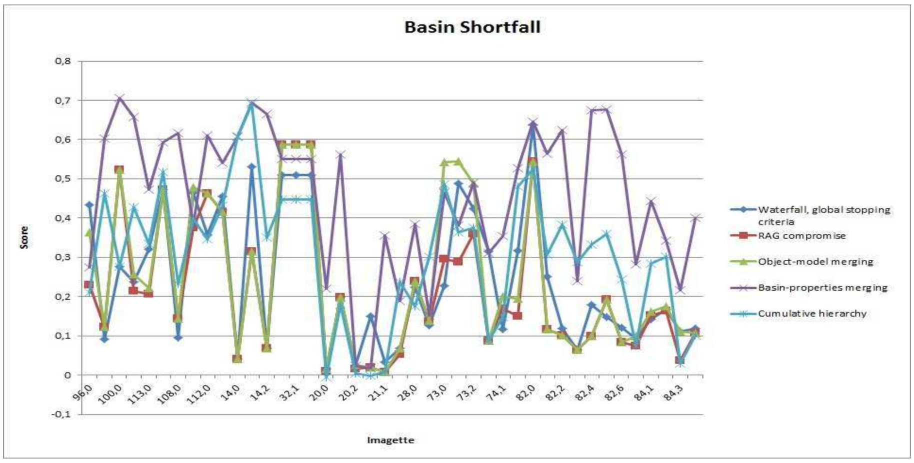
- Since the third is in fact a derivative of the first, we can logically count them as one.



(a)

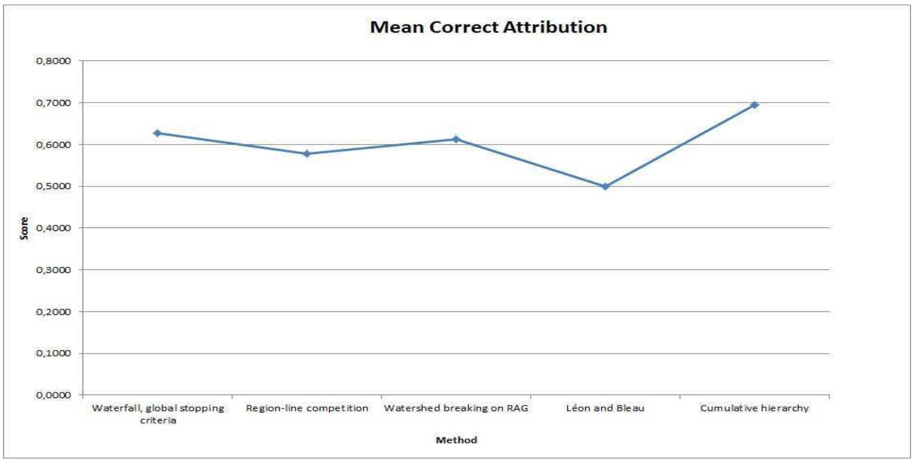


(b)

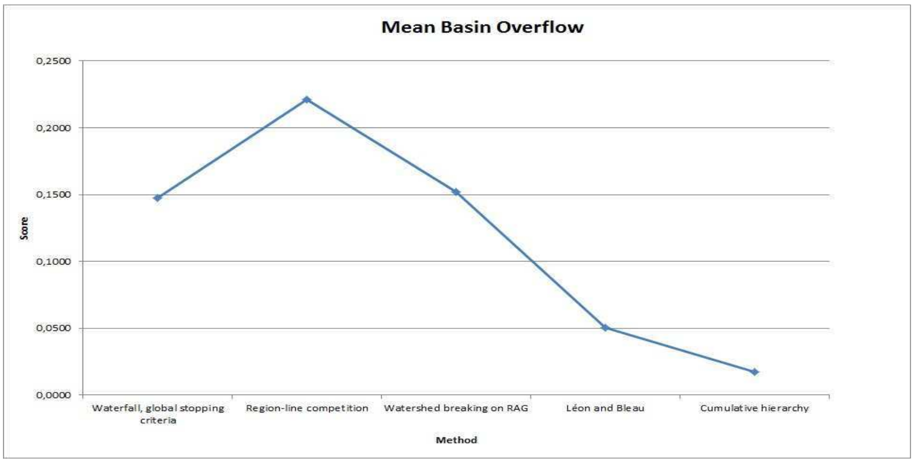


(c)

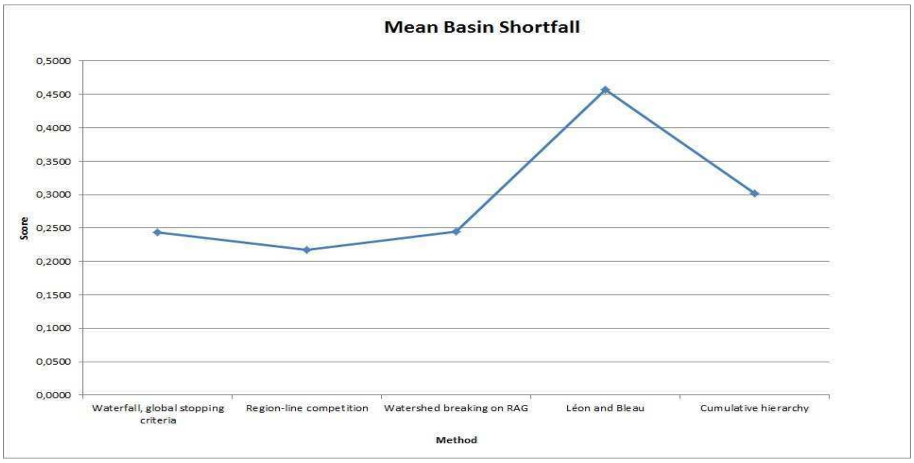
**Figure 4.13:** *Graphs of (a) Correct segmentation (b) basin overflow and (c) basin shortfall for the entire subimage data.*



(a)



(b)



(c)

**Figure 4.14:** *Trends in mean of (a) Correct segmentation (b) basin overflow and (c) basin shortfall.*

**Table 4.3:** *Mean and standard deviation of Basin Overflow by improvement algorithm.*

| Method | Waterfall, global stopping criteria | Region-line competition | Watershed breaking | Léon-Bleau | Cumulative hierarchy |
|---|---|---|---|---|---|
| Mean | 14.74% | 22.12% | 15.22% | 5.07% | 1.73% |
| Stdev | 17.34% | 17.77% | 15.31% | 9.97% | 2.37% |

**Table 4.4:** *Mean and standard deviation of Basin Shortfall by improvement algorithm.*

| Method | Waterfall, global stopping criteria | Region-line competition | Watershed breaking | Léon-Bleau | Cumulative hierarchy |
|---|---|---|---|---|---|
| Mean | 24.41% | 21.71% | 24.46% | 45.66% | 30.23% |
| Stdev | 17.35% | 17.73% | 19.32% | 18.42% | 17.21% |

- Among the remaining two, cumulative hierarchy is found to consistently have the largest $CA$, while also having the advantage of not requiring an initialization at all.
- The ad hoc method of RAG merging based on local similarity criterion worked surprisingly well as long as it was initialized so that the decision was among the few fragments of cells in an aggregate.

**Table 4.5:** *Mean Correct Attribution with and without initialization.*

| Method | Region-line competition | Watershed breaking | Léon-Bleau |
|---|---|---|---|
| With init. | 57.81% | 61.34% | 49.89% |
| Without init. | 38.33% | 38.33% | 38.62% |

As can be seen, the Basin Shortfall is around 62%, i.e., that percentage of every cell is not included in the region that corresponds to it. At the amount of over-segmentaion that exists in the images, region-merging processes completely fail if not given a little headstart through a swamping process. Even though cumulative hierarchy only fares twice as better as region-mergind methods without initialization for the $BS$ score, the latter is in the same ball-park as all the region-merging methods with initialization, so that result is not too unexpected.

The second segmentation quality criterion, which is not represented in these graphics, has a story to tell. Even the object-model and region-line-competition algorithms suffer more than cumulative hierarchy on visual inspection when it comes to merging regions from different cells together. This is possibly because of the implied thresholds, and because of the fact that the criteria implemented represented characteristics of segmented connected components / regions and not of the image

**Table 4.6:** *Mean Basin Overflow with and without initialization.*

| Method | Region-line competition | Watershed breaking | Léon-Bleau |
|---|---|---|---|
| With init. | 22.12% | 15.22% | 5.07% |
| Without init. | 0.12% | 0.12% | 0.13% |

**Table 4.7:** *Mean Basin Shortfall with and without initialization.*

| Method | Region-line competition | Watershed breaking | Léon-Bleau |
|---|---|---|---|
| With init. | 21.71% | 24.46% | 45.66% |
| Without init. | 62.27% | 62.27% | 61.91% |

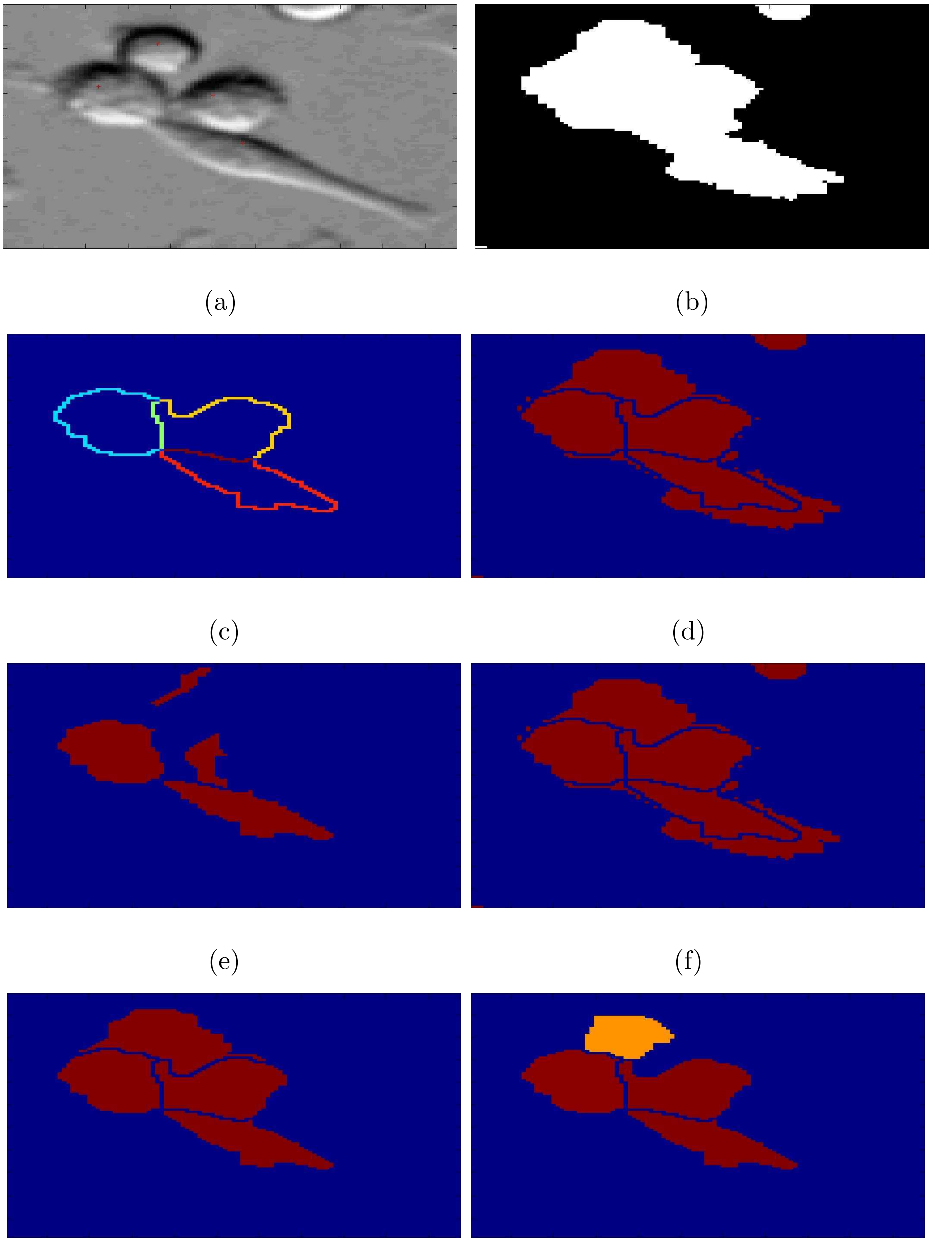
topography, something logically much simpler and intrinsically present. The reader can gather an idea about the subjective $SQE$ criterion from Fig. 4.15, which has been selected to offer a honest appreciation of the success and indeed the lack of in the use of the various merging algorithms.

Thus the result of the exercise is the selection of the cumulative hierarchy method for our segmentation application. The result of this segmentation will now be passed on to the phase of the process that computes various characteristics that define the various cellular classes from these connected components.

## 4.6 Conclusion

In this chapter we detailed the second and arguably the most arduous phase of our work, i.e., segmentation of cellular images into *imagettes* containing individual cells. The cells in our images are not easily discernible from the background because of very similar grey levels, and are clustered together in cellular agglomerates. A first contribution to this image segmentation problem was presented in Chapter 2 where we presented a method to binarise, i.e., segment into two classes, the pixels from an image using an image transform and a classification-based thresholding technique. In this chapter, we first treated the subject of image segmentation and its application to cellular imaging, and then presented the *watershed transform* as the method of choice in our application of image segmentation. We also outlined a simple yet discriminatory method for assessing the quality of a segmentation in Section 3.3 that would help us to present a comparative analysis of the various methods implemented in later sections.

Our contributions to the state of the art in watershed segmentation are described



(a) (b)

(c) (d)

(e) (f)

(g) (h)

**Figure 4.15:** *Merging and hierarchical algorithms' visual comparison: a) original image selection with detected centres superimposed, b) the corresponding binarisation, c) the RLRAG used to give non-hierarchical methods a head start; and now the results of the merging methods: d) Criteria-based merging on the RLRAG, e) Bleau and Léon's Significance-of-basins method, f) Our extension of the previous method i.e. Region-line competition merging; and finally g) the result of the Waterfall algorithm under global stopping criteria, and h) our cumulative hierarchy or multi-scale marking.*

in two separate places, i.e., Chapter 3 Section 3.5 and Section 4.3. However, the reader will realize that the two form a holistic idea, the first principally being a description of the best input data for the second.

Chapter 3 Sections 3.4 and 3.5 discusses the ingredients necessary for the implementation of the watershed transform to a particular segmentation problem. First a minor simplification in the actual watershed algorithm is offered that consists of adding for each grey value pixels one by one to the priority queue instead of all pixels at that grey level at once, to eliminate the risk of mal-attribution of a pixel from one basin to another. However, algorithms are not the focus of this thesis and therefore apart from this simple modification we implement the Vincent and Soille algorithm in its originality. The real focus of this part of the chapter was the implementation of the watershed transform on various topographic reliefs obtained from the original greyscale image. These reliefs not only define the input data to the watershed, but veritable strategies for segmentation that quite often suffice for a correct and acceptable segmentation result for many authors as we saw in the previous chapter [71, 24, 20, 72, 219, 131, 183]. We prove that the use of a probabiltity function that describes membership scores to classes *cell* and *background* for each pixel in the image offers advantages over more traditional functions such as the gradient and the distance transform in terms of a more accurate representation of object shapes, a more well-defined separation between basin depths between the two classes, a desirable property for any post-processing region-merging operation as well as for the method we describe in 4.3 due to the fact that the deeper basins belonging to cells persist longer in the described hierarchical traversal thereby reducing the chance of a background basin becoming part of a cell.

The current chapter deals with our strategy of segmentation of cellular images, the presentation of a novel method of information preservation within a hierarchical watershed framework, and a comparison with several classes of segmentation-refining methods employing region merging. It is worthy of note that all five methods described in section 4.4 were implemented and the implementations are described in detail wherever we differed from the original proposition, in particular the watershed-breaking algorithm 4.4.3 and the basin-line-competition algorithm 4.4.1.B. An original *RAG* construction algorithm is also described in section 4.4.0.C, which is computationally much less expensive than its alternatives in the literature. Unlike region merging however, our method called *cumulative hierarchy* prevents over-segmentation rather than being a post-segmentation operation. Our bottom-up implementation proceeds thus: we modify the topographic image function at each iteration through geodesic reconstruction by erosion by itself increased by a constant level $h$ and this modified image function is segmented using a watershed transform. This process allowed us the latitude of not defining strict watershed marker that would not permit the possible of an eventual merging into centre-containing-basins those basins that lie in the formers' vicinity and potentially belong to cells. This, coupled with the characteristic of the probability-based topographic function that well-discriminates cell and background pixels, producing desired image regions corresponding to the extents of the marked-basins at each level of the hierarchy. At the same time control, an *accumulator* image permits us to preserve the regions marked at inferior levels of the hierarchy but are absorbed into the background as the hierarchy tends toward the global minimum of the image, hence the name of the method. This method strictly solely uses *mathematical morphology*, and due to this quality, integrates seamlessly and elegantly into a framework that uses only morphological *greyscale image reconstruction* first by erosion and then by dilation and a watershed transform between the two. The reason for the selection of this method, however, is that it was able to translate this elegance of intuition into practical segmentation quality. Cumulative hierarchy manifests a noticeable improvement in segmentation, allowing a *Correct Attribution* (for definitions refer to Chapter 3 Section 3.3) of about 70% while the three region-merging methods remain in the region of around $50\% - 60\%$. Incidently, even with global critera to stop it, the waterfall scores better than the latter at around 62%. Thus cumulative hiearchy not only improves in theory on classical waterfall while allowing the suppleness of not discounting shallow but desireable basins, but also practically by correctly segmenting around 8% more of the image in our rather less-than-ideal case. As detailed results in section 4.5 show, cumulative hiearchy is by a large margin superior in *Basin Overflow* as well, while being inferior in its *Basin Shortfall* score, something due to the topographic function which very differently measures grey levels on pixels strictly inside of cells and those on the fringes and in the background. In the future, we would like to improve on this criterion as well, by introducing a basin-size measure to the currently basin-depth mechanism of the hierarchy. We also hope to publish this method internationally.

The work described in this chapter allows us to obtain as accurately as possible the shape, size and grey level information of every cell detected in Chapter 2 in the form of *connected components*. The shape and size information can now be used to formulate shape features, and the grey level information texture features, for the purpose of classifying the cells, which forms the subject of the work described in the following chapter.

# 5

# Classification of cells

## 5.1 Introduction

The knowledge of metastasic progress, i.e., the knowledge of how many cells are in each phase of metastasis is the final objective of this work. Once we have arrived at this stage in the process, we possess the information on the number of cells, their coordinates on an image, and their connected component in the form of both the bounding box that defines the imagette as well as the binary mask representing the connected component coordinates. Fig. 5.1 shows an example of this information on the subimage 0032.1 in continuity with the previous example. This example shows well both the accuracy and the shortcomings in the segmentation algorithm. This information will now be exploited to extract discriminatory knowledge about the shape, size, grey level and texture of the cell represented by the imagette to arrive at a decision for each cell whether it is a spread cell, a smooth round or transitory cell, or a blebbing cell. Let us briefly introduce this classification process and its application in our context.

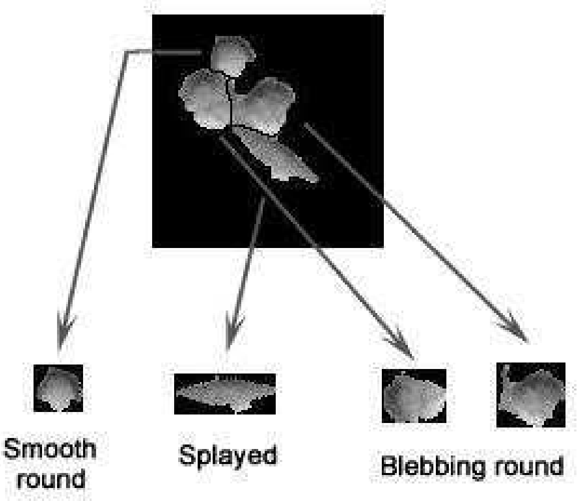


**Figure 5.1:** *Example of data issued from cell image segmentation.*

**Table 5.1:** *Percentage distribution of the* 3 *cell classes in our* 14*-image sample dataset, labeled by an expert.*

| % | Smooth round | Blebbing round | Spread Spread | Smooth +Blebbing | Blebbing/ (Smth+Bleb) | Smooth/ (Smth+Bleb) |
|---|---|---|---|---|---|---|
| Mean count | 59.00 | 143.71 | 326.00 | 202.71 | - | - |
| Mean % | 11.67 | 30.38 | 57.95 | 42.05 | 73.39 | 26.61 |

Statistical classification is the problem of identifying the sub-population to which new observations belong, on the basis of a training set of data containing observations whose sub-population is known. Therefore these classifications will show a variable behaviour which can be studied by statistics. Thus the requirement is that new individual items are placed into groups based on quantitative information on one or more measurements, traits or characteristics, etc. and based on the training set in which previously decided groupings are already established. The problem here may be contrasted with that for cluster analysis, where the problem is to analyze a single data-set and decide how and whether the observations in the data-set can be divided into groups. In certain terminology, particularly that of machine learning, the classification problem is known as *supervised* learning, while clustering is known as *unsupervised* learning. A learning classifier is able to learn based on a sample. The data-set used for training consists of information $x$ and $y$ for each data-point, where $x$ denotes what is generally a vector of observed characteristics for the data-item and $y$ denotes a group-label. The label $y$ can take only a finite number of values.

The classification problem can be stated as follows: given training data $\{(x_1, y_1), \ldots, (x_n, y_n)\}$ produce a rule (or "classifier") $h$, such that $h(x)$ can be evaluated for any possible value of $x$ (not just those included in the training data) and such that the group attributed to any new observation, specifically $\hat{y} = h(x)$, is as close as possible to the true group label $y$. For the training data-set, the true labels $y_i$ are known but will not necessarily match their in-sample approximations $\hat{y}_i = h(x_i)$. For new observations, the true labels $y_j$ are unknown, but it is a prime target for the classification procedure that the approximation $\hat{y_j} = h(x_j) \approx y_j$ as well as possible, where the quality of this approximation needs to be judged on the basis of the statistical or probabilistic properties of the overall population from which future observations will be drawn. Given a classification rule, a *classification test* is the result of applying the rule to a finite sample of the initial data set.

Microscopy is a consequential application for pattern recognition that presents many diverse problems and image modalities [26, 168, 196]. When pattern recognition has been used, the tendency is to tailor the image descriptors as well as the classification algorithm to a specific type of imaging problem. Many methods for analyzing images have been proposed during the last few years. Beil et al. [10] proposed a dual approach to structural texture analysis for microscopic cell images, in which textures are composed of primitives and can be described by arrangement of regions and lines. Thiran and Macq [204] presented an automatic recognition method based on shape and size analysis for the observed cells in cancerous tissues and provided an evaluation method for scoring the images to be classified. A Biopsy Analysis Support System (*BASS*) was introduced by Schnorrenberg et al. [176] to detect the nuclei of breast cancer based on staining intensity and the number of stained nuclei. Esgiar et al. [56] developed an algorithm to identify cancerous colon mucosa using six texture features. Yi et al. [231] developed a computer assisted differential diagnosis system based on syntactic structure analysis, which utilized $k$-nearest-neighbor ($KNN$) algorithm with parameters selected from the Voronoi Diagram ($VD$) and the Minimum Spanning Tree ($MST$).

The classification of segmented cells is understood to be the assignment of one of the following labels to it: **spread** ($S$), **smooth round** ($T$), or **blebbing** ($B$). This process can be viewed as an information refinement procedure: we begin with the cut-out imagette of the cell - the information, i.e. pixel values, is copious but not directly relevant, so we transform the imagette into a set of measures that describe the cell and compress the information of all cells into a table, but since these measures are not all equally important for decision making, we select the most discriminatory ones, and finally the classifier reduces the information for each such measure, called a *feature vector*, into a *decision label.*

To summarize, the decision process consists of:

1. **Feature extraction**:
   calculation of the set of measures that describe a cell.
2. **Feature or variable selection**:
   selection of the most discriminatory features.
3. **Classification**:
   learning and predicting a decision label for the cell.

The rest of the chapter follows this three-step chain of action. A few comments are appropriate at this point, and we treat them in some detail in the following discussion.

We will present our classification results on the 14-image dataset. Cells have already been labeled by experts for this dataset, and therefore we have adopted a supervised approach to classification - we will learn the best configuration of the selected variables for each of the three labels, and predict the label for any new data presented.

In lieu of variable selection, we initially tried dimensionality reduction through $PCA$ [95]. One purpose for Principal Component Analysis is to transform the features to orthogonality. Another is to try to find the linear combinations of features which have the maximum variation. Figure 5.2 produces a bi-plot of the variables

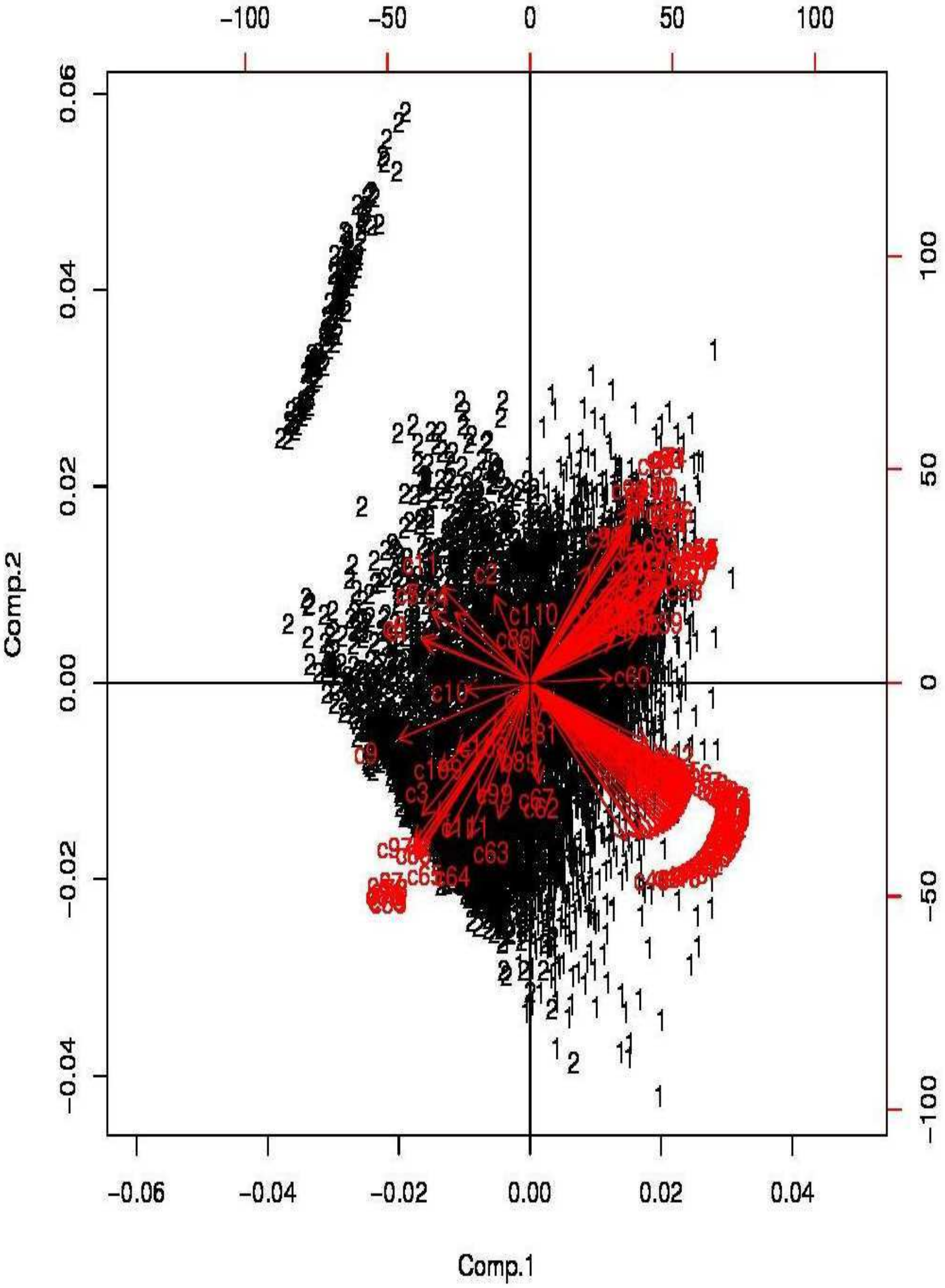


**Figure 5.2:** *Second principal component versus first principal component for all cell examples.*

and of the observations in a factorial plan. The variable scores in the PCA can be drawn as bi-plot arrows that point in the direction of increasing values for that variable. Although PCA failed to achieve the best combination of variables for classification, and hence the results are not cited, the plot reveals a wealth of information, which ultimately set us on our classification strategy: we observe one large cluster of variables $c_i$ in the bottom-right quadrant, and a smaller one in the top-right quadrant, and a number of other variables scattered in different and indeed opposing directions. This means that a number of features all group to define one class, labeled 1, pertaining to spread cells. Since most cells labeled 1 are on the right and most labeled 2 on the left, the deduction is obvious: it is easier to classify cells into the classes spread and round, and a subset of the features, i.e. the large clusters on the left, is clearly more favorable to distinguish them. Looking closely at those features we found most features describing cell morphology/shape grouped within it. Therefore the classification strategy is the following:

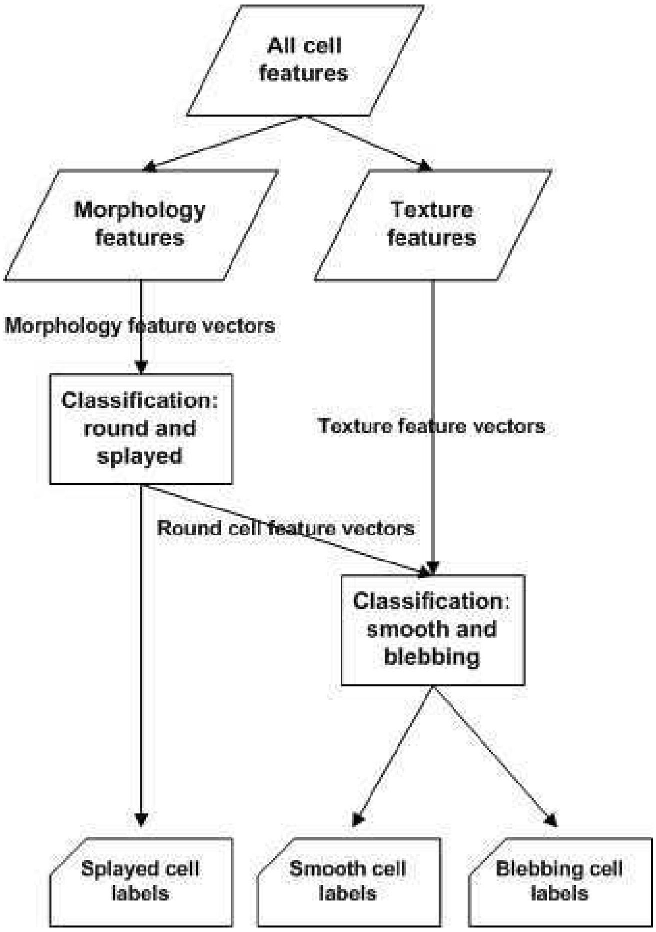


**Figure 5.3:** *Flow diagram representing data movement along the hierarchical classification strategy.*

1. Separate all round cells and spread cells using features pertaining to cell morphology.
2. Within the classification space of round cells, separate smooth round cells from blebbing round cells using features pertaining to the texture within the imagette.

We understand that the most discriminatory combination of all kinds of cellular features, morphological or textural, can be arrived at through meticulous analysis, but for the purpose of these experiments, this morphology versus texture division appeared to be a logical choice. Hence once the selected feature vectors are retained, we divide them into morphology and texture sets and follow the classification strategy outlined in Fig. 5.3, i.e. first spread cells and round cells, and then within the round cells, smooth and blebbing cells.

## 5.2 Definition of cellular characteristics

Although the image plane is the carrier of various patterns, pixels themselves are not normally used directly as inputs to machine learning algorithms. Instead, image content is derived through computation of numerical values that represent quantitative measures of various pixel patterns [76] [87]. These numerical features of the image are based on different algorithms that extract a wide variety of patterns present in the image, such as edges, color [205], textures [61], shapes [143], histograms [165] etc.

In order to distinguish between *simple round*, *blebbing* and *spread* metastasic cells, we need to extract distinguishing features of each from the image and compute new variables that concentrate information to separate classes. Such feature set has to consist of features leading to large between-class distance and small within-class variance in the feature vector space, i.e. the set of features should discriminate between different classes as well as possible. An additional requirement is robustness, so that the results can be reproduced for new independently collected material. Raw images cannot be used directly as features due to high variations in morphology which are coupled with arbitrary rotations and scales and because the raw images contain large amount of data, but relatively little information. This is the aim of feature extraction to transform the input data into a reduced set of features that extract the relevant information from the input data. Rodenacker and Bengtsson [171] have surveyed a large collection of content descriptors for the analysis of grayscale microscopy images. They differentiated feature types into four major categories:

*intensity*, *size and shape*, *texture*, and *structure*. Following the concept introduced in [171], the feature extraction process can be expressed in terms of the definition of the zone of measurement, i.e. the *imagette*, and a non-scalar set of measurements on the latter, i.e. the *feature vector*. This is the process generally followed in this work.

The selection of the features for classification was based on the visual differences between the classes of cells as are commonly used by technicians for manual microscopic diagnosis, and the feature selection used by other cytological studies. The chosen features can be grouped into two categories: morphological features, and texture features including those pertaining to grey level.

Morphological features express the overall size and shape of the cell without taking the grey levels represented inside of the cell into account. In other words, these features are computed only from the binary region representing the cell which was obtained by the preceding segmentation process and the actual gray level image is not needed. Since absolute measurements, such as orientation, absolute coordinates, and absolute dimension are inutile for the application, we have to choose features which are invariant under translation, changes in scale, and rotation.

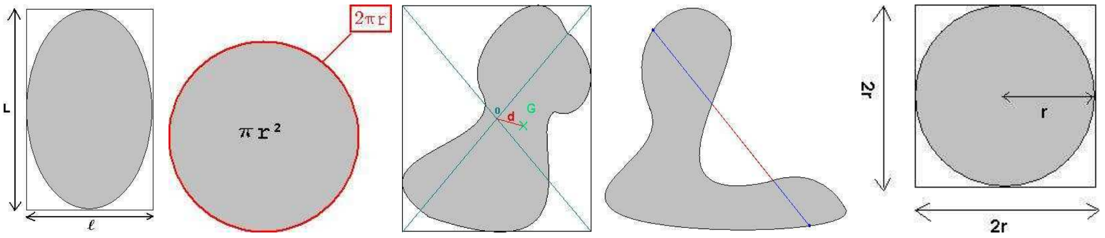
Grey level features are based only on the absolute value of the intensity measurements in the image. In grey level based measurements, the spatial positions of the pixels are not taken into account, only the information retained in an histogram of the image.

## 5.2.1 Morphology features

Although all these features are closely related to the size of the object relative to the estimated cell radius, they all describe different aspects of the shape.

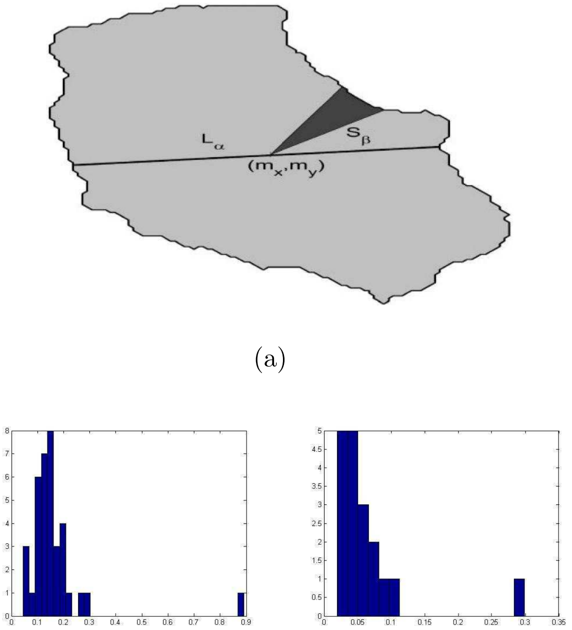
### 5.2.1.A Connected component region and contour properties

We also used a set of common region (segmented $CC$) properties to describe the morphology and texture characteristics of the cells. For general texture description, the maximum, minimum mean value, and standard deviation of the intensity in the segmented cell area were used. Moreover, we used some elementary morphological descriptions, such as the lengths of the longest axis $L$ and the shortest axis $l$, the ratio $L/l$ (Fig. 5.4 a), the area $s$ of the cell, the perimeter $p$ of the cell, and the compactness of the cell, which is calculated as compactness $= p^2/(4p \times s)$ (Fig. 5.4 b). If the perimeter of the minimum convex shape is $p_c$, then the roughness is roughness $= p/p_c$. In all, we extracted 7 general intensity and shape features for each segmented cell region: length of long axis, length of short axis, long axis / short axis, area, perimeter, compactness, and perimeter of convex hull of the cell.



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

**Figure 5.4:** *Illustration of some morphological descriptors: a) major and minor axes, b) compactness, c) Centroid and centre-of-gravity disparity, d) convexity, and e) ratio cell-to-inscribed-ellipse (round cell w.r.t circle shown).*



(a)

(b) (c)

**Figure 5.5:** *The computation of the shape descriptor using ratio length and ratio area: a) cellular region, central axis, and fan bin, b) the* 36*-dimensional ratio length feature, c)the* 18*-dimensional ratio area feature.*

*Contour irregularity:*

The depressions and protrusions on the cell wall potentially define the degree of blebbing on a cell. Hence we study the countour, i.e., the edge of the region representing the cell, in a greater detail. The contour can be represented by a sequence of sampling boundary points $\{p_0 p_1 p_2 ... p_{j-w} ... p_j ... p_{n-1}\}$ with $p_t = p_{t+n}$ for $t = ... -101....$

The curvature at a boundary point $p_j = (x_j y_j)$ can be approximated by the differentiation of two successive tangent values in window $w$ [98]:

$$d_j = tan^{-1}\frac{y_j - y_{j-w}}{x_j - x_{j-w}} - tan^{-1}\frac{y_{j-1} - y_{j-1-w}}{x_{j-1} - x_{j-1-w}}$$

Since the morphology of a cell could be quite convoluted and fine in its detail, we chose $n =$ cell circumference. Then we define contour irregularity as

$$\text{contour-irreg} = \Sigma_{j=0}^{n-1} |d_j - d_{j-1}| \tag{5.1}$$

*Centroid and region centre-of-gravity distance:*
We know the position of the centroid of the region matter and it is easy to calculate the position of the center of the imagette. We can therefore calculate the distance between the center of the sticker and the centroid. This feature is shown in Fig. 5.4 c.

$$\text{cent-cog-dist} = \sqrt{(x_G - x_d)^2 + (y_G - y_d)^2} \tag{5.2}$$

For a perfect round, the two points obviously overlap and the distance is zero.

*Region convexity:*
In a convex figure, when connecting two points of the contour, the connecting line segment does not exit the figure. The round, for example, is a convex figure. It would be far too long to consider all pairs of contour points. We therefore choose a random point on the contour and see what happens when the links to all other points of contour. For each line segment, we look at the proportion that goes “outside” of the binary mask of the connected component. While averaging all the results obtained, we have a score related to the convexity. For a convex connected component or $CC$, this score will be zero. Refer to Fig. 5.4 d. for an illustration.

$$\text{conv} = \text{mean}(\frac{\text{number of non-CC pixels}}{\text{chord length}}) \tag{5.3}$$

*Region within inscribed ellipse:*
The correlation with an ellipse inscribed within the imagette provides a coarse clue of the morphology of the $CC$ contained in it. The less crooked the cell and the fuller it is, the closer this feature will it be to 1. The ratio of number of cell pixels in the ellipse are therefore counted, as shown in Fig. 5.4 e.

*Phenotype shape binning descriptor:*
Ideally, if we precisely define the boundary of each cell, the 3 different phenotypes

of metastasic cells have more obvious differences in morphology rather than texture. Because the shape information provided in the region property features is inexact, we developed 2 additional kinds of morphological descriptors as our problem-specific features. One feature is based on the ratio length of the central axis projection, and the other feature is the area distribution over each equal sector.

Inside the $CC$ the value of $f(x, y)$ equals 1 when the pixel $(x, y)$ is located in the cell area, otherwise it is 0. The centroid of the cellular area $(m_x, m_y)$ is obtained. Centered at the centroid, we get a series of the central axis as the line $L_\alpha$, shown in Fig. 5.5. The central projection along $L_\alpha$ denotes the length of the axis. The equation of $L_\alpha$ is based on the angle a of the axis and the centroid coordinate $(m_x, m_y)$. The ratio length of the central projection is defined as the length of $L_\alpha$ divided by the perimeter of the cellular contour.

$$r_{L_\alpha} = \frac{1}{p} \int f(x, y), \tag{5.4}$$

where $p$ is the same with the perimeter calculated in region property. For each different angle $a$, the ratio length for the central axis is calculated. The angles are evenly sampled with 36 different values to derive a 36-dimensional ratio length feature that represents the shape of the cellular boundary.

The second morphological descriptor is based on the distribution of sector areas. As shown in Fig. 5.5, a sweeping “fan” bin $S_\beta$ centered at the centroid is denoted. The ratio area is defined as the area of the bin to the area of the entire cellular region.

$$r_{S_\beta} = \frac{\int\int f(x, y)_{(x,y) \in S_\beta}}{\int\int f(x, y)} \tag{5.5}$$

The entire cellular region is angle-evenly partitioned into 18 sectors. Hence, the ratio area feature is constructed by the ratios of each sector. Figure 5.5 b,c gives the computed shape features represented by the histograms, where the $x$-axis denotes the index of the sector and the value of the $y$-axis represents the percentage of area of the cell region or the length of the cell boundary falling into this sector. These 2 morphological descriptors are scale and translation invariant but rotation variant. To achieve rotation independence, the calculated ratio length and ratio area are sorted by value.

### 5.2.1.B Zernike moments

Zernike moments are part of the geometrical moment's general theory. They were introduced initially by F. Zernike. Zernike moments are built on a set of orthogonal polynomials which allow construction of orthogonal base given by Eq. 5.6.

$$V_{n,m}(x,y) = V_{n,m}(\rho,\theta) = R_{n,m}(\rho).\exp(j.m.\theta) \tag{5.6}$$

where

$$R_{n,m}(\rho) = \Sigma_{k=|m|}^{n} \frac{(-1)^{(n-k)/2}.(n+k)!}{\frac{(n-k)}{2}!.\frac{(k+m)}{2}!.\frac{(k-m)}{2}!} \tag{5.7}$$

$$\rho = \sqrt{x^2+y^2}, \quad \theta = argtan(\frac{y}{x}) \tag{5.8}$$

with $n \geq 0$, $m \neq 0$, $m < n$, $n - m < n$ $n - k$ even.

$R_{n,m}(\rho)$ are the orthogonal radial polynomials, $n$ is the order of the moment and $m$ the repetition factor (the smoothness of the required details) at this order. $\rho$ and $\theta$ are respectively the radius and the angle of the function's point under consideration.

Simply speaking, the Zernike moments features of an image are calculated based on the particular weighted averages of the intensity values. They are generated with the basis functions of Zernike polynomials. As classical image features, Zernike moments have wide applications. Here, we give a brief description for calculating Zernike moments features for each cell.

1. Calculate the center of mass for each cell polygon image and redefine the cell pixels based on this center.

2. Compute the radius for each cell, and define the average of the radii as $R$.

3. Map the pixel $(x, y)$ of the cell image to a unit circle and obtain the projected pixel as $(x', y')$.

Because the Zernike moments polynomials are defined over a circle of radius 1, only the pixels $(x', y')$ within the unit circle will be used to calculate Zernike moments. Finally, 49 Zernike moments features are computed.

### 5.2.2 Texture Features

Textural features aim to quantify the overall local density variability inside the object of interest. Textural features are more complex, more difficult to define in a unique, robust and reproducible way, and they are more difficult to understand intuitively. Moreover, it is often difficult to visualize textural features and relate specific feature values to appearance of cells. The difficulty to relate the textural features to visually perceived changes in object structure and appearance is the large disadvantage of these features, especially of the co-occurrence and run-length types. However, these features also proved to be one of the most useful ones in many cytological studies [171].

#### 5.2.2.A First order statistics

These are very basic representations of the texture of a connected component, but as we have seen in Chapter 4, are still used as means for measuring textural homogeneity of an image region, in that case for a merging decision. What we use here are the intensity or grey level values, their mean over the connected component, their minimum and maximum over the connected component, and their standard-deviation over the connected component. In addition, we believe that the hierarchy level at which a cell region disappeared from the watershed hierarchy during the cumulative hierarchy algorithm is an indication of how textured the cell was, since more texture meant higher partial-membership assignment values and a slower fusion and later extinction of the region by swamping, and have included this level as an elementary texture measure.

#### 5.2.2.B Co-occurrence Matrix Features

Co-occurrence matrix is a spatial-dependent matrix representation of the image which estimates the probability that a pixel $I(k,l)$ has intensity $i$ and a pixel $I(m,n)$ has intensity $j$ [28]. Supposing the probability depends only on a certain spatial relation $r$ between a pixel with intensity $i$ and a pixel with intensity $j$, then the information about the relation $r$ is recorded in the co-occurrence matrix $C_r$ with dimensions corresponding to the number of intensity levels in the image. The spatial relation $r$ can be represented by displacement vector $D$ which is often expressed as distance $d$ and angle $\theta$ . Let $L_X = \{1, 2, ..., N_X\}$ denote the horizontal spatial domain of the analyzed image with resolution $N_X \times N_X$, and let $L_X = \{1, 2, ..., N_X\}$ denote the vertical spatial domain and $G = \{1, 2, ..., N_G\}$ be the set of $N_G$ quantized grey levels. The input image $I$ is represented as $I : L_Y \times L_X$. Then the co-occurrence matrix $C$ of dimensions $N_G \times N_G$ for displacement vector $D = [d_1, d_2]$ is defined as originally proposed in [82].

$$\begin{aligned} C(i,j,D) = \texttt{freq}\{&((k,l),(m,n)) \in (L_Y \times L_X) \times (L_Y \times L_X) \\ &|k - m = d_1, l - n = d_2 I(k,l) = i, I(m,n) = j\} \end{aligned} \tag{5.9}$$

where `freq` denotes the number of elements in the set.

The co-occurrence matrix can be seen as an accumulator matrix to which 1 is added at $C(i,j)$ if a co-occurrence specified by intensities $i$ and $j$ and the spatial relation given by $D$ is found. The co-occurrence matrix defined by 5.9 is not symmetrical. The symmetrical co-occurrence matrix can be obtained by using absolute values in the distance conditions: $|k - m| = d_1$ and $|l - n| = d_2$. Then the ordering of values in the pixel pairs is not considered and $C(i,j,D) = C(j,i,D)$.

Since the texture in cells is directionally homogeneous, we can calculate the co-occurrence matrix using only one displacement vector. However, in order to reduce any irrelevant directional dependencies, we also apply the displacement vector with a rotation of 90° and accumulate the results to the co-occurrence matrix. The following displacement vectors are used: $D = [0, d]$ which corresponds to angle $\theta = 90°$ and $D = [d, 0]$ which corresponds to angle $\theta = 90°$.

The parameters controlling the extraction of the co-occurrence matrix are, in addition to the displacement vector $D$, also the number of quantized gray levels $N_G$, which determines the size of the co-occurrence matrix, and the normalization method, which determines how the gray-scale values are scaled to the gray levels. The number of gray levels can, theoretically, be any number. However, for large numbers of levels, the co-occurrence matrix may become sparse with limited generalization properties. The normalization method may be, for example, histogram equalization or linear spread.

The co-occurrence matrix features can be calculated for any of the previously described transformed images as well as for the original extinction image. In our case, co-occurrence matrix is generated from the segmented *imagette*. The 13-feature set originally proposed in [82] is derived from the co-occurrence matrix and is used as below.

We evaluate the co-occurrence matrix features for lengths of the displacement vector $d = 1$, and a number of gray levels $N_G = 256$. The co-occurrence matrix is calculated in the four adjacent pixels i.e. with $\theta = 0°, 90°, 135°, 180°$ and the result averaged. The extracted co-occurrence features were as follows: angular second moment, contrast, correlation, sum of squares, inverse difference moment, sum average, sum variance, sum entropy, entropy, difference variance, difference entropy, information measures of correlation, and maximal correlation coefficient.

#### 5.2.2.C Gabor Features

In the spatial domain, a $2D$ Gabor filter is a Gaussian kernel function modulated by a sinusoidal plane wave. The Gabor filters are self-similar: all filters can be generated from one mother wavelet by dilation and rotation.

The Gabor wavelet features were developed by Manjunath and Ma [132] and are formed by a set of multiscale and multiorientation coefficients to describe texture variations in an image. The Gabor wavelet features have been used as the texture signature for numerous image analysis applications, such as image retrieval, segmentation, and recognition [27, 43]. Gabor filters are directly related to Gabor wavelets, since they can be designed for a number of dilations and rotations. However, in general, expansion is not applied for Gabor wavelets, since this requires computation of bi-orthogonal wavelets, which may be very time-consuming. Therefore, usually, a *filter bank* consisting of Gabor filters with various scales and rotations is created. The filters are convolved with the signal, resulting in a so-called Gabor space. This process is closely related to processes in the primary visual cortex [42]. As defined by Daugman, the 2-dimensional complex-value Gabor function is a plane waverestricted by a Gaussian envelope. After conducting the Gabor wavelet transformation on the cell image, the real and imaginary parts of the transformation coefficients $C_R$, $C_I$ can be obtained. The magnitude of the transformed coefficients $C = \sqrt{C_R^2 + C_I^2}$ is used as the Gabor vector.

The discrete wavelet transformation ($DWT$) has been adopted to investigate image characteristics in both scale and frequency domains. In our work, we applied an important wavelets technique, the Gabor wavelet to extract phenotype texture. Because the transformed coefficients are computed based on pixels, this procedure derives the magnitudes for each pixel in the image. In the texture feature extraction method [132], the statistics, such as mean $\mu$ and standard deviation $\sigma$ of these magnitudes on the entire image, are calculated as the feature representation. Through changing the scales and orientations, a set of feature representations can be calculated, which provide rich texture signatures in the frequency domain. In our experiments, 4 scales and 6 orientations are used to compute a 70-dimensional feature $(\mu_{0,0}, \sigma_{0,0}, \mu_{0,1}, ...\mu_{4,6}, \sigma_{4,6})$ for each segmented cell.

### 5.2.3 What does the data look like?

The cell features can be categorized into 2 categories: morphological features and textural features. These are produced by the aforementioned processes i.e. intensity measurements, discrete texture (Haralick) calculations, Gabor and Zernike characteristics. Recapitulating very briefly, morphological features estimate the size, shape, and boundary variations of the nuclear object of an image using measurements such as area, $x-$centroid, $y-$centroid, mean radius, max radius, var radius, sphericity, eccentricity, inertia shape, compactness and boundary variation. Photometric features give estimations of absolute intensity. The discrete texture features are based on segmentation of the object into regions. Gabor-based features are obtained by applying a set of Gabor filters with 6 equidistant orientations and 4 resolutions, resulting in 70 means and standard deviations filtered images. Zernike moments have proved to be superior in terms of their feature representation capability and low noise sensitivity; Zernike features describe shape in terms global estimation of differences of phase and amplitude. The list of features is given in Table 5.2.

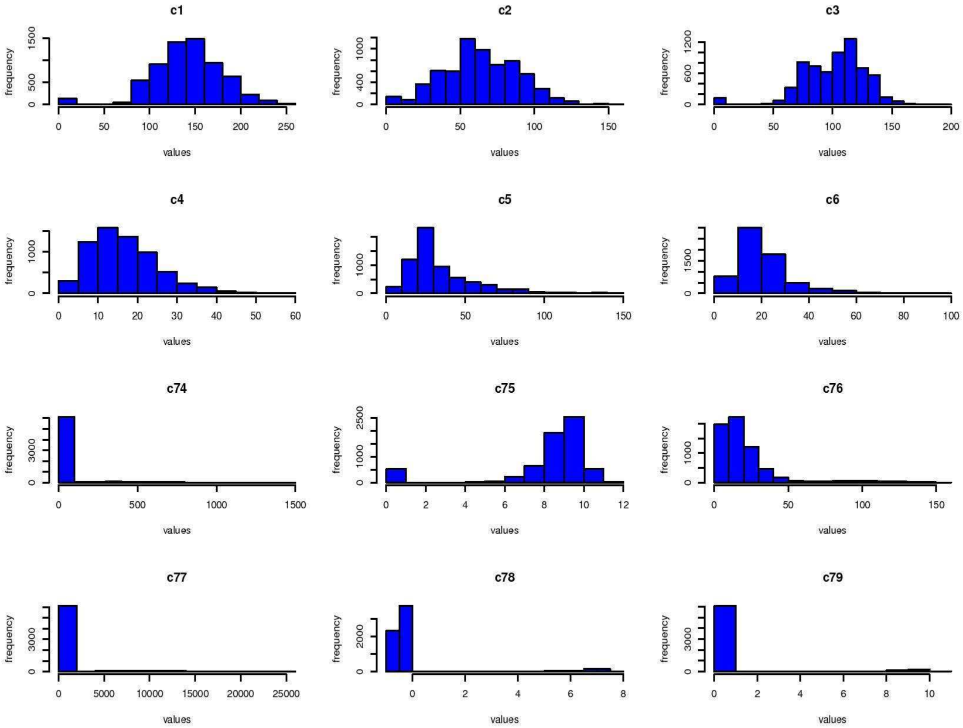


**Figure 5.6:** *Histograms of the first few features for the complete set of examples.*

Figure 5.6 gives the histogram of the values for some characteristics, both morphological and textural. Their distributions do not reveals a universal structure in the data. For example, the first feature or characteristic $c1$ is Gaussian, $c3$ is bimodal, the last few distributions are very narrowly-clumped and so forth. However, a natural division of the distribution into two or indeed three distinct classes for the same data examples is not present, and shows that the need of classification is warranted.

| Feature Type | Features' list | Brief description |
|---|---|---|
| Morphology: region simple | $c_6 \cdots c_{13}$ | estimate size, shape, and boundary variations of the nuclear object of an image |
| Morphology: region and contour | $c_{14} \cdots c_{17}$ | centre centroid diff., chord-convexity, inscribed ellipse correlation, contour irregularity |
| Morphology: Zernike | $c_{87} \cdots c_{133}$ | Zernike coefficients |
| Morphology: "Phenotype binning" | $c_{18} \cdots c_{72}$ | spatial histogramming |
| Texture: $1^{st}$ order | $c_1 \cdots c_5$ | segmentation region extinction level, and mean, min, max and standard deviation of grey level |
| Texture: co-occurrence | $c_{73} \cdots c_{86}$ | Haralick's measures: entropy, energy, contrast, correlation, homogeneity $\cdots$ |
| Texture: Gabor | $c_{134} \cdots c_{202}$ | mean and standard derivation for each Gabor-transform image |

**Table 5.2:** *Cell parameters as extracted variables.*

## 5.3 Feature Selection

The features described above might contain redundancies. Some of them might also be non-discriminative (i.e., not able to contribute to the classification task). Feature reduction has shown to increase classification accuracy as well as speed up a classifier [90], [26].

In machine learning literature, feature selection methods are traditionally divided into filter methods (or simply "filters"), which perform feature selection independent of any particular inducer, and wrapper methods (or "wrappers"), which try to optimize the feature set for a given classifier (refer to Guyon [78] for a detailed review). More recently a third category, the embedded feature selection methods, has been added to this system. This chapter is organized according to this system.

The selection of variables by filters is based on a metric calculated directly on the data. This method does not take into account the performance of the classifier in the selection of variables, unlike the other two approaches developed in the future. Metrics are usually based on the notions of dependence (linear correlation), distance (Mahalanobis distances, etc.). Or information (entropy) or on statistical tests and allow for an order of relevance of the variables. Guyon and Wlodzislaw [53] [22] address comprehensively the various metrics recognized and widely used in the selection of variables and their advantages.

A key problem is to discover the most relevant variables, or features, among the tens of thousands of parallel measurements in a particular experiment. This is referred to as feature selection. For feature selection to be principled, one needs to decide exactly what it means for a feature to be "relevant". This thesis considers relevance from a statistical viewpoint, as a measure of statistical dependence on a given target variable. The *p*redictive features are those that allow an accurate predictive model. The most predictive features may not always be the most relevant ones from a biological perspective, since the predictive power of a given feature may depend on measurement noise rather than biological properties.

We had several questions concerning feature selection which we could not resolve within available literature. The most important were the following:

- What does "relevance" mean? When is a feature relevant to a given target variable? Are there perhaps several perspectives on relevance, and do we need to choose a particular perspective in a subjective fashion?
- What is the relation between a good predictive model and the features relevant to that model?
- What types of feature selection methods are feasible given the limited sample sizes we have access to?

### 5.3.1 Statistical data models

In a statistical data model, we think about experimental systems as statistical distributions. The model is "statistical" in that, when experiments are repeated, there is some random variation in the measurements which we cannot explain by the experimental design. We might be able to control some of the "nuisance" variables by experimental design but this is not always possible. Moreover, many factors that influence the cell migration are probably unknown to us since our knowledge of biology is incomplete, and these are of course impossible to control. Also, the measurements themselves may be more or less corrupted with noise from various physical or chemical factors. Therefore, one will inevitably observe variation that cannot be explained. In a statistical perspective, we often speak about any variation that cannot be explained by the chosen model as noise. However, it should be understood that this does not imply that the variation is truly random. Much of the variation one observes between individuals in biology is probably deterministic, and could in principle be explained if our knowledge of biology was more complete.

In order to choose a subset of features from the morphology and texture feature sets, Stepwise Discriminant Analysis ($SDA$) [101] was used.

The goal of stepwise discriminant analysis is to sequentially identify those variables (features) that maximize a criterion which describes their ability to separate classes from one another while at the same time keeping the individual classes as tightly clustered as possible. The criterion used is Wilks' $\Lambda$ which is defined as

$$\Lambda = \frac{\det(W(c))}{\det(T(c))}, \tag{5.10}$$

where $c = (c_1, c_2, \ldots, c_p)^T$ is a vector of the features that are currently included in the system,

$$W(i,j) = \sum_{q=1}^{Q} \sum_{t=1}^{n_q} (c_{iqt} - \bar{c}_{iq})(c_{jqt} - \bar{c}_{jq}), \tag{5.11}$$

is the matrix of within-groups sums of squares and cross products for the features under consideration, and

$$T(i,j) = \sum_{q=1}^{Q} \sum_{t=1}^{n_q} (c_{iqt} - \bar{c}_i)(c_{jqt} - \bar{c}_j). \tag{5.12}$$

is the matrix of total sums of squares and cross products. $Q$ is the number of classes, $n_q$ is the number of samples in class $q$, $x_{iqt}$ is the value of feature $i$ for sample $t$ of class $q$, $\bar{c}_{iq}$ is the mean of feature $i$ over class $q$, and $\bar{c}_i$ is the mean of feature $i$ over all classes.

Low values of $\Lambda$ indicate features that better discriminate the classes. To accommodate the stepwise nature of the process, the partial $\Lambda$ statistic is used. This statistic describes the increase in the discrimination ability of a system after adding a new feature, $c_{p+1}$

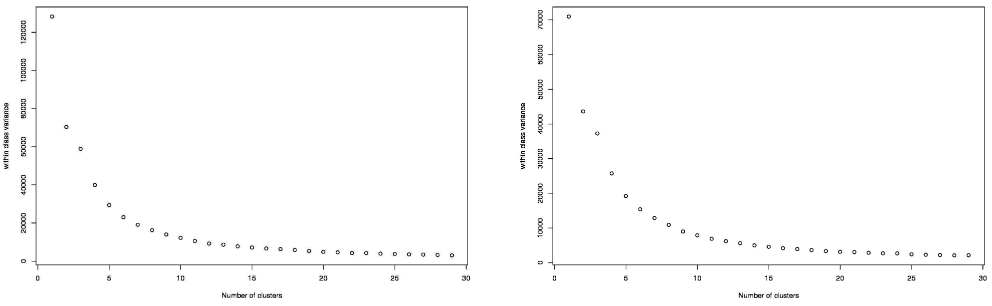
$$\Lambda(c_{p+1} \cdot x) = \frac{\Lambda([c, c_{p+1}])}{\Lambda(x)}. \tag{5.13}$$

To facilitate the ability to decide whether adding a new feature to the system will increase discrimination significantly, Wilks' partial-$\Lambda$ is converted to an $F$-statistic for which it is possible to assign a level of statistical significance: what is the probability, given the null hypothesis that there is no separation between groups, that one would obtain a value larger than

$$F = \left(\frac{n-q-p}{q-1}\right)\frac{1-\Lambda(c_{p+1}\cdot c)}{\Lambda(c_{p+1}\cdot x)} \tag{5.14}$$

where $n$ is the number of data samples in all classes, $p$ is the number of features currently in the analysis, and $Q$ is the number of classes. Large values of $F$ indicate better discrimination for a particular feature. This version of the $F$-statistic is used to decide whether feature $x_{p+1}$ should be entered into the system.

The process of stepwise discriminant analysis can be referred to in algorithmic details in [93].



(a) (b)

**Figure 5.7:** *The within-class difference* $\Lambda$ *is minimized as variables are eliminated, for morphology and texture features respectively.*

| Feature type | Features index in respective table |
|---|---|
| Morphology | $c_{70}, c_{1}, c_{5}, c_{42}, c_{103}, c_{8}, c_{67}, c_{11}, c_{36}, c_{46}, c_{102}, c_{91}, c_{9}, c_{78}, c_{79}, c_{112}, c_{97}, c_{6}, c_{4}, c_{12}, c_{62}, c_{23}, c_{14}, c_{2}, c_{55}, c_{66}, c_{48}, c_{82}, c_{85}, c_{105}, c_{94}, c_{92}, c_{71}, c_{109}, c_{29}, c_{104}, c_{93}, c_{107}, c_{10}, c_{106}, c_{95}$ |
| Texture | $c_{93}, c_{10}, c_{76}, c_{102}, c_{72}, c_{3}, c_{44}, c_{6}, c_{8}, c_{128}, c_{121}, c_{133}, c_{98}, c_{87}, c_{56}, c_{91}, c_{39}, c_{97}, c_{132}, c_{69}, c_{136}, c_{101}, c_{62}, c_{127}, c_{90}, c_{79}, c_{109}, c_{12}, c_{68}, c_{49}, c_{48}, c_{130}, c_{105}, c_{96}, c_{131}, c_{115}, c_{111}, c_{117}, c_{116}, c_{118}, c_{99}, c_{35}, c_{11}, c_{24}, c_{30}, c_{124}, c_{122}$ |

**Table 5.3:** *List of selected features.*

Table 5.3 lists the variables selected for morphology and texture features. Note that the new subscripts represent the indices in either dataset. Fig 5.7 details the $SDA$ variable selection process. The feature with the lowest $F$-statistic value that also corresponds to a significance level ($p$) greater than an assigned threshold ($p = 0.10$ i.e. a 90% confidence level) is removed from the list of features. That is, the most informative variable is eliminated. The $W$ and $T$ matrices are re-calculated. The degree-of-freedon $Df$ of the model therefore decreases at every step. The two rightmost columns describe the decreasing information in the statistical model through Wilk's $\Lambda$. At the end, 42 variables out of 114 are selected from among morphology variables, and 47 variables out of 89 are selected from among texture ones.

The fewer and more discriminatory features are now extracted from the initial morphology and texture feature tables into similar tables containing only the selected features. At this point we re-express $c_i$ as $\frac{c_i - \text{mean}(c)}{\text{std-dev}(c)}$. We want to do this because

1. Numerical stability is enhanced when all variables are on a similar scale.
2. variables of similar magnitude are easier to compare. For instance $c_i = 5,78$ is easier to parse for a classifier than $c_i = 0,000000578$
3. A change of units might aid interpretability.

One rather thorough approach to scaling is to convert all the variables to standard units (mean 0 and variance 1). This is called *centering* and *reduction* of the data. Once centeres and reduced, the selected variables are passed onto the classification step.

## 5.4 Classifying the selected features

Classification is the problem of the prediction of a categorical response $Y$ given a feature vector $X$. Let $(X, Y)$ denote the pair of a feature $X$ and class $Y$ for a unit drawn at random from the population. In the framework of a statistical model, the class prior for a class $r$ is denoted as $\pi_r = P(Y = r)$, and the conditional density of feature $x$ given $Y = r$ is denoted as $P(X = x|r)$ or simply $p(x|r)$. Given a probability space and two events $r$ and $x$, the conditional probability of $r$ given $x$ is denoted as $p(r|x)$. The posterior probability of an event is the conditional probability given the observation. The posterior probability of class $r$ given the realization of the random variable $X = x$ is given by Bayes formula:

$$p(x|r) = P(Y = r|X = x) = \frac{\pi_r p(x|r)}{\sum_{k=1}^{R} \pi_k p(x|k)}. \tag{5.15}$$

If we consider a classifier to be optimal if it minimizes the classification error, then the problem of allocation of future cases is optimized using Bayes rule.
Here, the true state is $r = 1, \dots, R$ and we can choose any of those classes ($k = 1, \dots, R$). The optimal rule is to simply choose the class with the highest posterior probability. Much work has been done in the approximation of the conditional distribution $p(r|x)$.

It is generally the case that the probability density $p(x) = f(x|Y = r)$ is not known. Many classification methods have been developed using both heuristic approaches and based on formal statistical models, including tree-based methods (CART, random forests), Neural Nets, Support Vector Machines, Kernel and Nearest Neighbor methods. Methods based on formal statistical models or which are at least statistically motivated include Linear Discriminant Analysis ($LDA$), Quadratic Discriminant Analysis ($QDA$), Logistic regression, Estimated Bayes methods, and Naive Bayes. It is important to note that some problems are better suited for different classification methods. The lack of a dominating method for every problem requires us to be able to employ all the tools at our disposal when approaching a classification problem to determine which one works best for our problem of interest. The process of selecting the best classifier may suffer from the problem of selecting one that is over-trained, working great on the data set that it was trained on, but poorly on an independent data set. A popular solution is to use cross validation in order to get unbiased estimates of the classifier's performance. Hastie *e*t al. suggested that if there is enough data, divide the data into three data sets: a training, validation, and test set [85], sampling in the proportions of 40%, 30%, and 30% in order to make the training, validation and test sets, respectively. The training set is used to estimate the parameters of a classifier. The validation set is used to get estimates of our trained classifier's performance, using the parameters estimated from the training set. The test set is reserved for the very end when we have chosen the best classifier (choosing one based on the estimated performances on the validation set) and want to get a final unbiased estimate of its performance on an independent set. We expect that the performance on the test set will deteriorate.
If we were to look at 10 classifiers that had the exact same theoretical performance, one will come out as best in the validation set out of all of them by chance, which of course will be an overestimate of its true performance. Then, when applied to the test set, its estimated performance would regress to the mean.

Many classification methods exist and have varying degrees of success. While some of them generally behave with better accuracy, none is best in every situation.

It is wise to try a number of classification procedures for a given problem in order to see which provides the best results for the problem at hand. We now present an overview of some of the procedures that we use on our data. It is by no means exhaustive nor all-inclusive. The methods were implemented in the R statistical package [198].

We start with data that are classified into several groups, and want a rule that will allow us to predict the group to which a new data value will belong. In the language of Ripley [170], our interest is in supervised classification.

### 5.4.1 Discriminant Analysis classification

Discriminant analysis is a method used in statistics, pattern recognition and machine learning to find a linear combination of features which characterize or separate two or more classes of objects or events. The resulting combination may be used as a linear classifier for partitioning the data, or, more commonly, for dimensionality reduction before later classification. In case the variables are linearly separable, the method is known as Linear Discriminant Analysis or $LDA$ [62].

$LDA$ is closely related to $ANOVA$ (analysis of variance) and regression analysis, which also attempt to express one dependent variable as a linear combination of other features or measurements [62, 137]. In the other two methods however, the dependent variable is a numerical quantity, while for $LDA$ it is a categorical variable (i.e. the class label). Logistic regression and probit regression are more similar to $LDA$, as they also explain a categorical variable. These other methods are preferable in applications where it is not reasonable to assume that the independent variables are normally distributed, which is a fundamental assumption of the $LDA$ method. $LDA$ works when the measurements made on independent variables for each observation are continuous quantities. When dealing with categorical independent variables, the equivalent technique is discriminant correspondence analysis.

For two classes in the data, consider a set of features or variables $\vec{x}$ for each sample of an object with known class $y$. This set of samples is called the *training set*. The classification problem is then to find a good predictor for the class $y$ of any sample of the same distribution (not necessarily from the training set) given only an observation $\vec{x}$. $LDA$ approaches the problem by assuming that the conditional probability density functions $p(\vec{x}|y = 0)$ and $p(\vec{x}|y = 1)$ are both normally distributed with mean and covariance parameters $(\vec{\mu}_0, \Sigma_{y=0})$ and $(\vec{\mu}_1, \Sigma_{y=1})$, respectively. Under this assumption, the Bayes optimal solution is to predict points as being from the second class if the ratio of the log-likelihoods is below some threshold T, so that;

$$(\vec{x}-\vec{\mu}_0)^T\Sigma_{y=0}^{-1}(\vec{x}-\vec{\mu}_0) + \ln|\Sigma_{y=0}| - (\vec{x}-\vec{\mu}_1)^T\Sigma_{y=1}^{-1}(\vec{x}-\vec{\mu}_1) - \ln|\Sigma_{y=1}| < T$$

Without any further assumptions, the resulting classifier is referred to as $QDA$ (*quadratic discriminant analysis*). LDA also makes the simplifying homoscedastic assumption (i.e. that the class covariances are identical, so $\Sigma_{y=0} = \Sigma_{y=1} = \Sigma$) and that the covariances have full rank. In this case, several terms cancel and the above decision criterion becomes a threshold on the dot product $\vec{w}\cdot\vec{x} < c$ for some threshold constant $c$, where $\vec{w} = \Sigma^{-1}(\vec{\mu}_1 - \vec{\mu}_0)$

This means that the criterion of an input $\vec{x}$ being in a class $y$ is purely a function of this linear combination of the known observations.

It is often useful to see this conclusion in geometrical terms: the criterion of an input $\vec{x}$ being in a class $y$ is purely a function of projection of multidimensional-space point $\vec{x}$ onto direction $\vec{w}$ . In other words, the observation belongs to y if corresponding $\vec{x}$ is located on a certain side of a hyperplane perpendicular to $\vec{w}$ . The location of the plane is defined by the threshold $c$.

As described earlier We use all classification algorithms in this description in two passes. First, feature vectors representing cells are classified using morphological features. This helps us to decide between spread cells and all round cells. In the future the same examples will be classified using texture variables, in order to distinguish between smooth round and blebbing round cells, hoping that the little more textural coarseness is sufficient to make the latter class stand out. The labels given by the expert (Table 5.1) are read as the prior probabilities.

Confusion matrices of the attribution of class labels by $QDA$ and $LDA$ are shown in Tables. 5.4 and 5.5 for morphology variables, and in Tables 5.6 and 5.7 for texture variables. As the reader can infer, despite the use of variable selection, the classes are still overlapping and confused. The amount of overlap can be seen in Figures 5.8 and 5.9. It is evident that morphological features, even though overlapping, are separable both linearly and quadratically, while texture parameters produce two clusters that are very intermixed. We hope to improve cluster compactness by the methods in the following section.

$QDA$ and $LDA$ appear to be complementary in their classification of spread and round cells using only morphology features. The first correctly classifies around 65% spread cells and around an equal percentage of round cells incorrectly, while $LDA$ too follows a roughly 60%-to-40% classification ratio but in the other direction i.e. it fares slightly better for the class of round cells. Fig. 5.8 illustrates the issue that the two clusters are intermingled in such a way that one is almost inside the other.

**Table 5.4:** *Confusion matrix for QDA on selected morphology features.*

| | | Predicted class | |
|---|---|---|---|
| | | **Spread** | **Round** |
| Actual class | **Spread** | 65.00% | 35.00% |
| | **Round** | 64.80% | 35.20% |

**Table 5.5:** *Confusion matrix for LDA on selected morphology features.*

| | | Predicted class | |
|---|---|---|---|
| | | **Spread** | **Round** |
| Actual class | **Spread** | 43.50% | 56.50% |
| | **Round** | 34.60% | 65.40% |

**Table 5.6:** *Confusion matrix for QDA on selected texture features.*

| | | Predicted class | |
|---|---|---|---|
| | | **Blebbing** | **Smooth** |
| Actual class | **Blebbing** | 80.70% | 19.30% |
| | **Smooth** | 26.20% | 73.80% |

**Table 5.7:** *Confusion matrix for LDA on selected texture features.*

| | | Predicted class | |
|---|---|---|---|
| | | **Blebbing** | **Smooth** |
| Actual class | **Blebbing** | 75.90% | 24.10% |
| | **Smooth** | 26.20% | 26.50% |

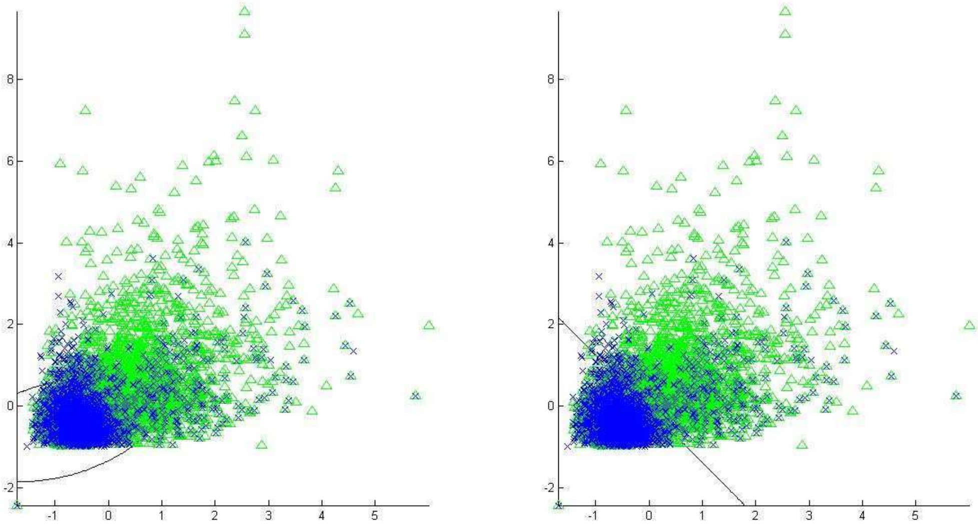


**Figure 5.8:** *Class distributions produced by QDA using only morphology features.*

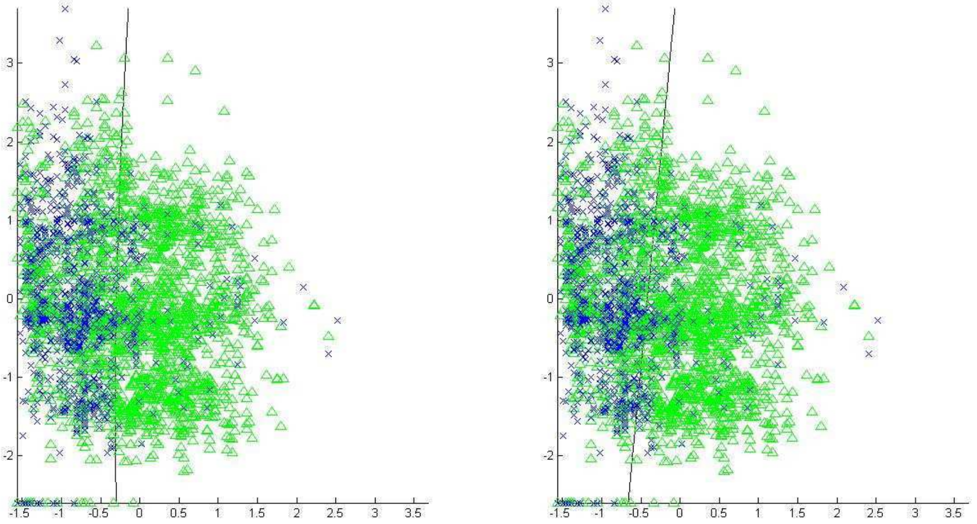


**Figure 5.9:** *Class distributions produced by LDA using only texture features.*

A linear separator cuts across the round cells' cluster, mal-classifying them.

Texture features prove better in discriminating smooth and blebbing round cells in both discriminant analyses. $QDA$ well-classifies more than 80% blebbing cells and about 74% smooth ones, and the trend carries over to $LDA$, which too correctly classifies around 75% blebbing and round cells each. Fig. 5.9 explains the phenomenon. The feature clusters are still very significantly overlap, but their centroids are further apart i.e. more of similar labels lie on each side of the separator than closer to it but on the other side.

These tables illustrate that Discriminant Analysis is not a very powerful tool in case of the features selected in the previous section, especially with regards to the classification based on morphological/shape parameters. We therefore tried one more classification method that is able to learn much more complex data models than a discriminant analysis hyperplane could be able to effectively separate. We attempt the same classification, with the same two-pass hierarchy using a neural network.

### 5.4.2 Artificial Neural Networks classification

Artificial Neural Networks ($ANN$) have a strong biological background. In the field of the brain cortex, local regulated and folded receptive field is the characteristic of the reflection of the brain. Based on this characteristic, this computational model was inspired, and for greater detail on the subject we refer the reader to [104]. An artificial neural network consists of an interconnected group of artificial neurons, and it processes information using a connectionist approach to computation. In most cases an $ANN$ is an adaptive system that changes its structure based on external or internal information that flows through the network during the learning phase. In an artificial neural network, simple artificial nodes, variously called "neurons" or "processing elements" ($PE$s), are connected together to form a network of nodes mimicking the biological neural networks. Modern neural networks are non-linear statistical data modeling tools. They are usually used to model complex relationships between inputs and outputs or to find patterns in data.

In modern software implementations of artificial neural networks, the approach inspired by biology has been largely abandoned for a more practical approach based on statistics and signal processing. In some of these systems, neural networks or parts of neural networks (such as artificial neurons) are used as components in larger systems that combine both adaptive and non-adaptive elements. While the more general approach of such adaptive systems is more suitable for real-world problem solving, it has far less to do with the traditional artificial intelligence connectionist models. What they do have in common, however, is the principle of non-linear, distributed, parallel and local processing and adaptation.

*Models*:
Neural network models are essentially simple mathematical models defining a function $f:X\rightarrow Y$ or a distribution over $X$ or both $X$ and $Y$, but sometimes models are also intimately associated with a particular learning algorithm or learning rule. A common use of the phrase *ANN* model really means the definition of a class of such functions (where members of the class are obtained by varying parameters, connection weights, or specifics of the architecture such as the number of neurons or their connectivity).

*Network function*:
The word network in the term 'artificial neural network' refers to the inter-connections between the neurons in the different layers of each system. An example system has three layers. The first layer has input neurons, which send data via synapses to the second layer of neurons, and then via more synapses to the third layer of output neurons. More complex systems will have more layers of neurons with some having increased layers of input neurons and output neurons. The synapses store parameters called "weights" that manipulate the data in the calculations.

An *ANN* is typically defined by three types of parameters: 1) The interconnection pattern between different layers of neurons, 2) The learning process for updating the weights of the interconnections, and 3) The activation function that converts a neuron's weighted input to its output activation.

Mathematically, a neuron's network function $f(x)$ is defined as a composition of other functions $g_i(x)$, which can further be defined as a composition of other functions. This can be conveniently represented as a network structure, with arrows depicting the dependencies between variables. A widely used type of composition is the nonlinear weighted sum, where $f(x)=K\left(\sum_i w_i g_i(x)\right)$ , where $K$(commonly referred to as the activation function) is some predefined function, such as the popular *sigmoid tangent* approximating but less brutal than, a step function. It will be convenient for the following to refer to a collection of functions $g_i$ as simply a vector $g=(g_1,g_2,\ldots,g_n)$.

Before classification, the image feature data were separated into distinct training and test sets in order to assess performance on images not seen by the classifier during training. Numbers of train/test images for each class are described in Table 5.8. After this separation, the training data were used to calculate the mean and variance of each feature. These values were then used to normalize the training data to have a mean of zero and a variance of one for each feature. The same mean and variance were then used to normalize the test data (the resulting means and variances for the test set therefore differed somewhat from zero and one respectively). The normalized training and test sets were used with the neural network classifier. Back-propagation neural networks ($BPNN$) were implemented using the R package neuralnet. Networks were configured with the number of inputs equal to the number of features being used, 10 hidden nodes, and 1 output nodes (with a sigmoid activation function). The learning rate was empirically chosen to be 0.2 at initialization, and the momentum was 0.9. The desired outputs of the network for each training sample were defined as 1 when corresponding to round cell class and 0 in the other case. The $BPNN$ was always trained using the single training data set defined above. After every third epoch of training, the sum of squared error was calculated for the stop data, where the error of the output node is defined as the difference between its expected and actual output values. Training of the network was terminated when the sum of squared error for the stop set reached a minimum. The performance of the network at the stopping point was measured using the corresponding evaluation set. The classification results summed to generate the confusion matrices in Tables 5.9 and 5.10 for both classification sub-problems. When measuring the performance of the network using the evaluation data, each sample was classified as belonging to the class corresponding to the largest output values.

| CLASS | | **Training** | **Test** | **Sub-problem class total** | **Class total** |
|---|---|---|---|---|---|
| **Spread** | | 2500 | 1167 | 3627 | 6463 |
| **Round** | **Blebbing** | 1500 | 511 | 2011 | 2836 |
| | **Smooth** | 500 | 325 | 825 | |

**Table 5.8:** *Learning and test distributions in the cell features database.*

**Table 5.9:** *Confusion matrix generated from the output of a back-propagation neural network for selected morphology features.*

| | | `Predicted class` | |
|---|---|---|---|
| | | **Spread** | **Round** |
| `Actual class` | **Spread** | 83.01% | 16.99% |
| | **Round** | 16.54% | 83.46% |

The BPNN was chosen as the second classifier because it is able to generate decision boundaries that are significantly more complex than the rectilinear and

**Table 5.10:** *Confusion matrix generated from the output of a back-propagation neural network for selected texture features.*

| | | Predicted class | |
|---|---|---|---|
| | | **Blebbing** | **Smooth** |
| `Actual class` | **Blebbing** | 67.16% | 32.84% |
| | **Smooth** | 11.88% | 88.12% |

conical boundaries of the discriminant analysis models. The disadvantage to the BPNN is that the ready interpretability of the classification tree is lost and those class separation boundaries are not known. It is not possible, for example, to easily determine which features are being used to discriminate which classes. In order to prevent over-training and therefore "memorization" of the training data, training was stopped when the sum of squared error value for the stop data was at a minimum. At this point, the evaluation data were applied to the network and the output node of the network with the largest value was defined as the classification result for each evaluation example. Results are shown in Tables 5.9 and 5.10. However, it is evident that the BPNN was still not able to well-discriminate blebbing cells from smooth round cells (correct calssification was around 67%). This is because of the fact that the feature selection method assumes linear correlation among the variables and the data possibly represents nonlinear dependances between variables that might be either noisy and nonlinear. Nevertheless,a neural network is a highly non-linear classifier, and is able to work with a large number of variables, so it was alimented with the two entire sets of variables. These results are shown in Tables 5.11 and 5.12, and as expected the BPNN was able to achieve superieor results. The average rate of correct classification for morphology features is around 90% for spread cells and around 80% for round cells. Within the *round* class, the average rate of correct classification for texture features is around 95% for blebbing cells and around 87% for smooth round cells. These results outclass both discriminant analyses in both levels of the classification hieararchy. Moreover, LDA and QDA both floundered for *blebbing* cells in particular and *round* cells in general. The neural network does not suffer from such lopsided classification. It can be concluded that the BPNN is an improvement over the classification tree in terms of its ability to classify the images in our context.

**Table 5.11:** *Confusion matrix generated from the output of a back-propagation neural network for all morphology features.*

| | | Predicted class | |
|---|---|---|---|
| | | Spread | Round |
| Actual class | Spread | 89.41% | 10.59% |
| | Round | 19.81% | 80.18% |

**Table 5.12:** *Confusion matrix generated from the output of a back-propagation neural network for all texture features.*

| | | Predicted class | |
|---|---|---|---|
| | | Blebbing | Smooth |
| Actual class | Blebbing | 94.88% | 5.12% |
| | Smooth | 12.49% | 87.51% |

## 5.5 Conclusion

In this chapter we presented an adapted strategy for classifying the cells obtained from image segmentation. The shape, size, contour, intensity, grey-level co-occurrence and other information is used to formulate representative characteristic features for the purpose of classifying the cells. A total of 203 features was thus obtained. Following exploratory experiments it was decided to split these features into morphological and textural feature tables, and use them to classify *spread versus round* cells and *blebbing versus smooth* cells respectively. As is usually normative in classification, the number of features was reduced through linear feature selection in order to filter out less discriminatory features. The selected features are first classified using discriminant analysis, and we achieve around 65%, 65%, 80% and 74% in the best of the two discriminant analysis cases for *spread*, *all round*, *blebbing round* and *smooth round* cells respectively. The classification based on morphologyy not being satisfactory, we decided to use a highly non-linear predictor, the neural network. A back-propagation neural network in its turn produced correct classification rates of around 90% and 80% for the *spread* versus *all round* case using morphological features, and around 95% and 87% for the *blebbing round* versus *smooth round* case using textural features, without prior feature selection, thereby proving to be the method of choice for classification in our application context.

# Conclusions and Perspectives

Let us begin by recalling the objectives of this work. How microenvironment patterns regulate cancer cell migration, proliferation or apoptosis can be studied by the use of phase-contrast microscopy to record the cellular responses over an extended period of time. Cells undergo *metastasis* i.e. they proliferate and migrate. Cancer cell migration is of two types: *mesenchymatic* or *amoeboid*, which translates to particular cellular morphologies, namely *spread* and *blebbing* respectively, along with the intermediate stage represented by the *smooth round* phenotype. The biologists which form out source of data are concerned with studying changes in migratory behaviour are through experimental observation associated with these morphologies of metastasic cells. In the context of numerical image processing, the goal was to determine how many and which cells are in each of the three phases of the metastasic process. This engendered sub-objectives of being able to recognize parts of the image as cells (*cell counting*), separating cells from the image background and from other cells (*cell segmentation*) in order to study their characteristics that represent the 3 phenotypes, then recognize the cells into differentiable categories (*cell classification*) according to their metastastic stage. This process determines the numbers and thus proportions of each of the 3 types of cells over an entire image, by first translating the overall problem into sub-problems concerning individual cells, and then re-combining those individual analyses into the global view of the process of metastasis.

The image acquisition procedure produces large *in vitro* cell images with specific issues, such as uneven illumination and low signal-to-noise ratios. Corrective preprocessing for mitigating these effects was first presented. Next the cell images were considered for cell detection and counting. We presented a new method for the automatic counting of cells. Counting results show that the proposed filter detects about 97% of the 66901 cells represented in the 142 image dataset, and commits few errors. This part of the work helps experts identify cells in less time and effort. The automatic count is only about 3% less than manual counts obtained by experts, given also that the uncertainty on manual is around 2%, which can be considered comparable. Then the maximum-likelyhood test devised for validating these results ensures that about 83% of the cells identified agree with reality.

For image segmentation, the choice of the method naturally led to the watershed transform. We have contributed through two improvements in its application: the topographic input function and the application strategy that limits over-segmentation. Our contributions to the state of the art in watershed segmentation are described in two separate places, i.e., Chapter 3 and Chapter 4. However, the reader would have realized that both form an holistic idea, the first principally being a description of the best input data for the second.

The focus of the latter part of Chapter 3 was the implementation of the watershed transform on various topographic reliefs obtained from the original greyscale image. These reliefs, such as distance or gradient maps, define not only the input data of the watershed but also dictate strategies for segmentation that often suffice for an acceptable segmentation result for many authors. We compared several topographic relief functions after having partially geodesically reconstructed by erosion them using a watershed segmentation. We found that the best correct segmentation score (*Correct Attribution*) of around 56% according to the segmentation-quality criterion defined earlier was offered by the fuzzy-probability map. We therefore proved that the use of a probability function that describes membership scores to classes *cell* and *background* for each pixel in the image offers advantages over more traditional topographic functions such as the gradient and the distance transform in terms of a more accurate representation of object shapes and more well-defined separation between basin depths between the cells and image background. As an outcome of this comparison the fuzzy-probability map was selected over competing topographic functions. The drawback associated of this topographic function is the splitting of the image into many small regions (so-called *over-segmentation*). We felt that the correct segmentation score could be improved well over 56% by mitigating this problem.

Chapter 4 reviewed several strategies to improve the segmentation of the cellular images by reducing over-segmentation. This consisted of the presentation of a novel method of information preservation within a hierarchical watershed framework, called *cumulative hierarchy* and its comparison with several classes of segmentation-refining methods employing region merging. It is worthy of note that all five methods described in section 4.4 were implemented and the implementations are described in detail wherever we differed from the original proposition, in particular the watershed-breaking algorithm 4.4.3 and the basin-line-competition algorithm 4.4.1.B. An original *region adjacency graph* construction algorithm is also described in section 4.4.0.C, which is computationally much less expensive than some of its alternatives in the literature. Unlike region merging however, cumulative hierarchy prevents over-segmentation rather than being a post-segmentation operation. It also allowed local image information to dictate the level of detail of the segmentation hierarchy. This method solely uses mathematical morphology and because of this integrates seamlessly and elegantly into a framework that uses only morphological *greyscale image reconstruction* first by erosion and then by dilation and a watershed transform between the two. Cumulative hierarchy manifested a noticeable improvement in segmentation, allowing a *Correct Attribution* of about 70% while the three region-merging methods remain in the region of around $50-60\%$. Incidently, even with global stopping critera, the waterfall scored better than the latter at around 62%. Thus cumulative hierarchy not only improves in theory on classical waterfall while allowing the suppleness of not discounting shallow but desireable basins, but also practically by correctly segmenting around 8% more of the image in a practical application context. The detailed results in section 4.5 show that cumulative hierarchy is by a large margin superior in *Basin Overflow* score as well, while being inferior in its *Basin Shortfall* score, something due to the topographic function in which pixel values of class *cell* are an amalgam of *cell-inside* and *cell-borders* and therefore do possess discontinuities that continue to over-segment cell boundary regions in many cases.

Automated region selection and cropping for classification can be exceptionally difficult and computationally expensive, especially when cells are highly confluent. Using cumulative hierarchy we were able to correctly segment about 70% cell pixels. The segmented regions issued from cumulative hierarchy are then extracted as *cellular connected components* in bounding boxes. Small values of *Basin Overflow* score mean that these connected components do not invade neighbouring cells, and the qualitative criterion shows that the connected components roughly do represent the shape of the cells they are associated with. These connected components were then used to extract relevant information for the classification work described in last chapter.

Chapter 5 takes the processing chain to its logical conclusion by classifying the cells obtained from image segmentation. The shape, size, contour, intensity, grey-level co-occurrence and other information is used to formulate representative characteristic features for the purpose of classifying the cells. A total of 203 features was thus obtained. Following exploratory experiments it was decided to split these features into morphological and textural features, and use them to classify *spread versus round* cells and then *blebbing versus smooth* cells respectively following a divide-and-conquer strategy. As is usually normative in classification, the number of features was reduced through feature selection in order to filter out less discriminatory features. A back-propagation neural network produced correct classification rates: respectively around 90% and 80% for the *spread* versus *all round* cells; respectively around 95% and 87% for the *blebbing round* versus *smooth round* cells.

To summarize, we have established a complete processing chain which permits, given a microscopic cellular image acquired by biologists, to obtain the number and proportion of cells in each class present over the image. This chain leans on a succession of steps - corrective pre-processing, cell detection, cell segmentation and finally cell classification, enabling us to find responses to the objectives initially outlined. Several communications and articles are in the course of preparation to valorize the methods developed and the results obtained at various stages of this work.

We have envisaged several extensions to the work. The limitation to the validation method for cell counting through hypothesis testing was that it supposed a rigid signal template and the absence of correlation between its columnar data. We are exploring an improvement that uses no prior knowledge and could offer in addition of a higher percentage of cell-validation.

We also plan to improve the cumulative hierarchy algorithm by limiting the rejection of minor peripheral cell regions into the background, by introducing a basin-size measure to the currently basin-depth mechanism of the hierarchy, and also by tweaking the topographic function to eliminate singularities towards the outer cell boundary.

Cell classification was limited in this study to a representative subset of the image dataset, only for which manual verification was available at the time. We now have to deploy cell classification, and as information tributary image segmentation as well, to the entire image dataset.

We believe that the cumulative hierarchy segmentation method can be easily adapted to a context wider than that of our application. We plan to extend it to images of different natures and formalize it as a general segmentation strategy.

# Bibliography

[1] Albini A. Tumor microenvironment, a dangerous society leading to cancer metastasis. from mechanisms to therapy and prevention. Cancer Metastasis Rev, 27(1):3–4, 2008 May.

[2] R. Adams and L. Bischof. Seeded region growing. IEEE Transactions on Pattern Analysis and Machine Intelligence, 16(6):641–647, 1994.

[3] P. S. U. Adiga and B. B. Chaudhuri. An efficient method based on watershed and rule-based merging for segmentation of $3-D$ histo-pathological images. Pattern Recognition, 34(7):1449–1458, 2001.

[4] O. Al-Kofahi, Radke R.J., Roysam B., and Banker G. Object-level analysis of changes in biomedical image sequences: Application to automated inspection of neurons in culture. In IEEE Transactions on Biomedical Engineering, volume 53(6), 2006.

[5] P. A. Arbelaez and L. D. Cohen. A metric approach to vector-valued image segmentation. International Journal of Computer Vision, 69(1):119–126, August 2006.

[6] G. Aubert and J. F Aujol J. F. Modeling very oscillating signals : Application to image processing. Research Report 4878, INRIA, France, July 2003.

[7] G. Ball and D. Hall. ISODATA, an iterative method of multivariate analysis and pattern classification. In IFIPS Congress, 1965.

[8] P. Bamford and B. C. Lovell. Method for accurate unsupervised cell nucleus segmentation. January 01 2001.

[9] L. Beaurepaire, J.-M. Ogier, and K. Chehdi. Identification aveugle d'images dégradées par un bruit additif ou multiplicatif. In GRETSI, Groupe d'Etudes du Traitement du Signal et des Images, 1997.

[10] M. Beil, T. Irinopoulou, J. Vassy, and G. Wolf. A dual approach to structural texture analysis in microscopic cell images. Computer Methods and Programs in Biomedicine, 48(3):211–219, 1995.

[11] J. Belien and H. van Ginkel et al. Confocal dna cytometry: A contour-based segmentation algorithm for automated three-dimensional image segmentation. Cytometry, 49(1):12–21, 2002.

[12] E. Bengtsson. Fifty years of attempts to automate screening for cervical cancer. Medical imaging technology, 17(3):203–210, 1999.

[13] K. A. Beningo, Dembo M., and Wang Y. L. Responses of fibroblasts to anchorage of dorsal extracellular matrix receptors. In Proc. Natl. Acad. Sci, 2004.

[14] R. Berry. Working in the digital darkroom. Astronomy, Aug. 1994.

[15] M. R. Berthold and J. Diamond. Boosting the performance of RBF networks with dynamic decay adjustment. In Advances in Neural Information Processing Systems, volume 7, pages 521–528. 1995.

[16] G. Berx, E. Raspé, G. Christofori, J.P. Thiery, and J.P. Sleeman. Pre-emting metastasis? recapitulation of morphogenetic processes in cancer. Clinical and Experimental Metastasis, 24(8):587–597, 2007.

[17] S. Beucher. Segmentation d'images et morphologie mathématique. PhD thesis, Ecole des Mines de Paris, jun 1990.

[18] S. Beucher. Watershed, hierarchical segmentation and waterfall algorithm. In Mathematical Morphology and its Applications to Image Processing, pages 69–76. 1994.

[19] S. Beucher and C. Lantuéjoul. Use of watersheds in contour detection. In International Workshop on Image Processing, pages 2.1–2.12, Rennes, September 1979. CCETT/IRISA.

[20] S. Beucher and Centre De Morphologie Mathmatique. The watershed transformation applied to image segmentation, June 28 1991.

[21] J. C. Bezdek. Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum Press, New York, 1981.

[22] J. Biesiada and W. Duch. Feature selection for high-dimensional data - A pearson redundancy based filter. In Computer Recognition Systems 2, volume 45, pages 242–249. 2008.

[23] C. Birchmeier and Birchmeier W. et al. Met, metastasis, motility and more. Nat. Rev. Mol. Cell. Biol, 4(12):915–25, 2003 Dec.

[24] A. Bleau and L. J. Leon. Watershed-based segmentation and region merging. Computer Vision and Image Understanding, 77(3):317–370, March 2000.

[25] A. Bleau and L. J. Léon. Watershed-based segmentation and region merging. Computer Vision and Image Understanding, 77(3):317–370, 2000.

[26] M. V. Boland, M. K. Markey, and R. F. Murphy. Automated recognition of patterns characteristic of subcellular structures in fluorescence microscopy images. Cytometry, 33(3):366–375, November 1998.

[27] A. C. Bovik, M. Clark, and W. S. Geisler. Multichannel texture analysis using localized spatial filters. IEEE Trans. Acoust., Speech, Signal Processing, 12(1):55–73, 1990.

[28] R. Boyle, V. Hlavac, and M. Sonka. Image processing, analysis, and machine vision. In Chapman and Hall, 1993.

[29] G. Cai, Lian J., Shapiro S. S., and Beacham D. A. Evaluation of endothelial cell migration with a novel in vitro assay system. Methods in Cell Science, 22(2-3):107–114, 2000.

[30] J. Canny. A computational approach to edge detection. IEEE Transactions on Pattern Analysis and Machine Intelligence, 8(6):679–698, nov 1986.

[31] A. Cartier-Michaud. Etude de l'influence du PAI-1 matriciel sur la régulation de la transition Mésenchymo-Amiboïde des cellules cancéreuses. PhD thesis, Université d'Evry, 2010.

[32] R. Castelló, J.M. Landete, F. España, C. Vázquez, C. Fuster, S.M. Almenar, L.A. Ramón, K.P. Radtke, and A. Estellés. Expression of plasminogen activator inhibitors type 1 and type 3 and urokinase plasminogen activator protein and mrna in breast cancer. Thrombosis Res, 120(5):753–762, 2007.

[33] G. J Chaitin. Meta Math! The Quest for Omega. Pantheon, 2005.

[34] T. F. Chan and L. A. Vese. Active contours without edges. IEEE Trans. Image Processing, 10(2):266–277, February 2001.

[35] A. Chauviere, Preziosi L., and Byrne L. A model of cell migration within the extracellular matrix based on a phenotypic switching mechanism. Mathematical medicine and biology : a journal of the IMA, 2009.

[36] K. Chehdi and M. Sabri. A new approach to identify the nature of the noise affecting an image. Acoustics, Speech, and Signal Processing, IEEE International Conference on, 3:285–288, 1992.

[37] I. Cheikhouhou, K. Djemal, and H. Maaref. Mass description for breast cancer recognition. In Image and Signal Processing, 4th International Conference, ICISP 2010, Trois-Rivières, QC, Canada, June 30-July 2, 2010. Proceedings, volume 6134, pages 576–584, 2010.

[38] C. Chettaoui, K.Djemal, and H. Maaref. 2d object description with discrete segments, 2005.

[39] S. Y. Chien, Y. W. Huang, and L. G. Chen. Predictive watershed: a fast watershed algorithm for video segmentation. IEEE Trans. Circuits and Systems for Video Technology, 13(5):453–461, May 2003.

[40] P. L. Correia and F. Pereira. Classification of video segmentation application scenarios. IEEE Trans. Circuits and Systems for Video Technology, 14(5):735–741, May 2004.

[41] Anoraganingrum D. Cell segmentation with median filter and mathematical morphology operation. In CIAP, pages 1043–1046, 1999.

[42] J. G. Daugman. Two-dimensional spectral analysis of cortical receptive field profiles. Vision Research, 20:847–856, 1980.

[43] J. G. Daugman. Complete discrete 2-D Gabor transforms by neural networks for image analysis and compression. IEEE Trans. Acoust., Speech, Signal Processing, 36(7):1169–1179, 1988.

[44] L. Davidson and R Keller. Basics of a light microscopy imaging system and its application in biology. Methods in Cellular Imaging. Oxford University Press, New York, 2001.

[45] J. Boutet de Monvel, Scarfone J. E., Le Calvez S., and Ulfendahl M. Image-adaptive deconvolution for three-dimensional deep biological imaging. Biophys J, 85(6):3991–4001, 2003.

[46] C. Ortiz de Solorzano, E. Garcia Rodriguez, and et al A. Jones. Segmentation of confocal microscope images of cell nuclei in thick tissue sections. Journal of Microscopy, 193(3):212–226, 1999.

[47] O. Debeir, P. Van Ham, R. Kiss, and C. Decaestecker. Tracking of migrating cells under phase-contrast video microscopy with combined mean-shift processes. IEEE Transactions on Medical Imaging, 24(6):697 – 711, June 2005.

[48] C. Decaestecker, Debeir O., Van Ham P., and Kiss R. Can anti-migratory drugs be screened in vitro? a review of 2d and 3d assays for the quantitative analysis of cell migration. Med Res Rev, 27(2):149–76, Mar 2007.

[49] R. Deriche. Using canny's criteria to derive a recursively implemented optimal edge detector. International Journal of Computer Vision, 1(2):167–187, 1987.

[50] Sébastien Derivaux, Sébastien Lefèvre, Cédric Wemmert, and Jerzy Korczak. Watershed segmentation of remotely sensed images based on a supervised fuzzy pixel classification. 2006.

[51] A. Desolneux, L. Moisan, and J.-M. Morel. Meaningful alignments. IJCV, 40(1):7–23, 2000.

[52] C. DiRuberto, Dempster A., Khan S., and Jarra B. Analysis of infected blood cell images using morphological operators. Image and Vision Computing, 20(2):133–146, 2002.

[53] W. Duch and R. Adamczak et al. A hybrid method for extraction of logical rules from data, 1998.

[54] J. C. Dunn. A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters. Journal of Cybernetics, 3:32–57, 1973.

[55] S. Eom, Kim S., Shin V., and Ahn B. Leukocyte segmentation in blood smear images using region-based active contours. In Advanced Concepts for Intelligent Vision Systems, volume 4179, pages 867–876. 2006.

[56] A. N. Esgiar, R. N. Gorgui-Naguib, and B. N. Sharif et al. Microscopic image analysis for quantitative measurement and feature identification of normal and

cancerous colonic mucosa. IEEE Transactions on Information Technology in Biomedicine, 2(3):197–203, 1998.

[57] Ambriz-Colin et al. Detection of biological cells in phase-contrast microscopy images. In MICAI '06, pages 68–77, 2006.

[58] M.P. Look et al. Pooled analysis of prognostic impact of urokinase-type plasminogen activator and its inhibitor pai-1 in 8377 breast cancer patients. J Natl Cancer Inst, 94(2):116–128, 2002.

[59] O. T. Fackler and Grosse R. Cell motility through plasma membrane blebbing. J Cell Biol, 181(6):879–84, 2008.

[60] G. M. Faustino, M. Gattass, S. Rehen, and C. de Lucena. Automatic embryonic stem cells detection and counting method in fluorescence microscopy images. In IEEE International Symposium on Biomedical Imaging: From Nano to Macro, 2009. ISBI '09, pages 799 – 802, 2009.

[61] C. J. S. Ferro and T. Warner. Scale and texture in digital image classification. Photogrammetric Engineering and Remote Sensing, 68(1):51–64, January 2002.

[62] R. A. Fisher. The use of multiple measurements in taxonomic problems. Annals of Eugenics, 7(7):179–188, 1936.

[63] P. Friedl and Wolf K. Tumour-cell invasion and migration: diversity and escape mechanisms. Nature Reviews, Cancer 3:362–374, May 2003.

[64] M. Frucci. A novel merging method in watershed segmentation. In ICVGIP, 2004.

[65] M. Frucci. Oversegmentation reduction by flooding regions and digging watershed lines. IJPRAI, 20(1), 2006.

[66] A. Garrido and de la Blanca N. P. Applying deformable templates for cell image segmentation. Pattern Recognition, 33(5):821–832, May 2000.

[67] H. G. Gauch. Scientific Method in Practice. Cambridge University Press, 2003.

[68] N. Gavert and A. Ben-Ze'ev. Epithelial-mesenchymal transition and the invasive potential of tumors. Trends Mol. Med, 14:199–209, 2008.

[69] J.B. Gibbs. Mechanism-based target identification and drug discovery in cancer research. Science, 287(5460), 2000.

[70] J. Gille, Meisner U., and Ehlers E.M. et al. Migration patternmorphology and viability of cells suspended in or sealed with fibrin glue: A histomorphologic study. Tissue and Cell, 37(5):339–348, 2005.

[71] R. C. Gonzalez, R. E. Woods, and S. L. Eddins. Digital Image processing using MATLAB. Pearson Prentice Hall, 2004.

[72] V. Grau, A. U. J. Mewes, M. Alcaniz, R. Kikinis, and S. K. Warfield. Improved watershed transform for medical image segmentation using prior information. IEEE Trans. Medical Imaging, 23(4):447–458, 2004.

[73] M. Grimaud. New measure of contrast: dynamics. In P. Gader, E. Dougherty, and J. Serra, editors, Image algebra and morphological image processing III, volume SPIE-1769, pages 292–305, jul 1992.

[74] B. Grosjean and L. Moisan. A-contrario detectability of spots in textured backgrounds. Journal of Mathematical Imaging and Vision, 33(3):313–337, 2009.

[75] L. Guigues, J. P. Cocquerez, and H. Men. Scale-sets image analysis. 2006.

[76] I. B. Gurevich and I. V. Koryabkina. Comparative analysis and classification of features for image models. 2006.

[77] T. Gustavsson, K. Althoff, and J. Degerman et al. Time-lapse microscopy and image processing for stem cell research-modeling cell migration. Proc. of SPIE, San Diego, USA, 10, 2003.

[78] I. Guyon and A. Elisseeff. An introduction to variable and feature selection. Journal of Machine Learning Research, 3:1157–1182, 2003.

[79] G. Hai, L. Weisi, X. Ping, and S. Wan-Chi. Marker-based image segmentation relying on disjoint set union. Sig. Proc: Image Comm, 21(2):100–112, 2006.

[80] D. Hanahan and Weinberg H. The hallmarks of cancer. Cell, 100(1):57–70, 2000.

[81] R. M. Haralick. Statistical and structural approaches to texture. Proc. IEEE, 67(5):786–804, May 1979.

[82] R. M. Haralick, K. Shanmugam, and I. Dinstein. Textural features for image classification. IEEE Trans. Systems, Man and Cybernetics, 3(6):610–621, November 1973.

[83] R. M. Haralick and L. G Shapiro. Survey: Image segmentation techniques. Computer Vision, Graphics and Image Processing, 29:100–132, 1985.

[84] R.M. Haralick and L. G Shapiro. Survey: image segmentation techniques. Comp. vis. Graph. Irn. Proc, 29:100–132, 1985.

[85] T. Hastie, R. Tibshirani, and J.H. Friedman. The Elements of Statistical Learning. Springer, New York, 2001.

[86] C. De Hauwer, Darro F., Camby I., Kiss R., Van Ham P., and Decaesteker C. In vitro motility evaluation of aggregated cancer cells by means of automatic image processing. Cytometry, 36(1):1–10, 21 Apr 1999.

[87] G. Heidemann. Unsupervised image categorization. Image and Vision Computing, 23(10):861–876, September 2005.

[88] B.P. Helmke, A.B. Rosen, and P.F Davies. Mapping mechanical strain of an endogenous cytoskeletal network in living endothelial cells. Biophys. J, 84:2691–2699, 2003.

[89] P. V. C. Hough. Methods and means to recognize complex patterns. U.S. Patent 3,069,654, 1962.

[90] K. Huang, M. Velliste, and R. F. Murphy. Feature reduction for improved recognition of subcellular location patterns in fluorescence microscope images. In spie, volume 4962, pages 307–318, 2003.

[91] D. Ioannou, W. Huda, and A. Laine. Circle recognition through a 2D hough transform and radius histogramming. Image Vision Comput, 17(1):15–26, 1999.

[92] R. Jarvis. On the identification of the convex hull of a finite set of points in the plane. Information Processing Letters, 2:18–21, 1973.

[93] R.I Jennrich. Statistical Methods for Digital Computers, volume 3, chapter Stepwise regression, pages 58–75. John Wiley & Sons, New York, 1977.

[94] F. Jänickem, A. Prechtl, C. Thomssen, N. Harbeck, C. Meisner, M. Untch, C.G. Sweep, H.K. Selbmann, H. Graeff, and M. Schmitt. Randomized adjuvant chemotherapy trial in high-risk, lymph node-negative breast cancer patients identified by urokinase-type plasminogen activator and plasminogen activator inhibitor type 1. J Natl Cancer Inst, 93(12):913–920, 2001.

[95] I. T. Jolliffe. Principal Component Analysis. 2002.

[96] N. N. Kachouie, L. Kang, and A. Khademhosseini. Arraycount, an algorithm for automatic cell counting in microwell arrays. BioTechniques, 47(3):10–16, 2009.

[97] M. Kass, A. Witkin, and T. Terzopoulos. Snakes: Active contour models. International Journal of Computer Vision, 1(4):321–331, 1987.

[98] H. Kauppinen, T. Seppanen, and M. Pietikainen. An experimental comparison of autoregressive and fourier-based descriptors in 2D shape classification. IEEE Trans. Pattern Analysis and Machine Intelligence, 17(2):201–207, February 1995.

[99] Z. Khan, T. Balch, and F. Dellaert. Multitarget tracking with split and merged measurements. In IEEE Computer Society Conference on Computer Vision and Pattern Recognition CVPR05, 2005.

[100] K. Kim, J. Jeon, W. W. Choi, P. Kim, and Y. S Ho. Automatic cell classification in human's peripheral blood images based on morphological image processing. In AI 2001: Advances in Artificial Intelligence, volume 2256 of Lecture Notes in Computer Science, pages 165–178. 2001.

[101] W.R Klecka. Quantitative Applications in the Social Sciences, volume 19, chapter Discriminant analysis. Sage University Paper, Beverly Hills and London, 1980.

[102] D. E. Knuth. The art of computer programming. 1973.

[103] A. Korzynska. Automatic counting of neural stem cells growing in cultures. In Computer Recognition Systems 2, volume 45 of Advances in Soft Computing, pages 604–612. 2007.

[104] D. Kriesel. A Brief Introduction to Neural Networks. 2007.

[105] A. Krtolica, C. Ortiz de Solorzano, S. Lockett, and J. Campisi. Quantification of epithelial cells in coculture with fibroblasts by fluorescence image analysis. Cytometry, 49:73–82, 2002.

[106] C. J. Kuo, S. F. Odeh, and M. C. Huang. Image segmentation with improved watershed algorithm and its FPGA implementation. IEEE, 2001.

[107] Y. H. Lai, P. W. Huang, and P. L Lin. An effective automated grading system for hcc in biopsy images. In Proceedings of the 11th WSEAS International Conference on Computers, pages 396–402, 2007.

[108] P. Lang, Yeow K., and Nichols A. et al. Cellular imaging in drug discovery. Nat. Rev. Drug Discov, 5(4):343–356, Apr 2006.

[109] C. Lantuéjoul. La squelettisation et son application aux mesures topologiques des mosaïques polycristallines. PhD thesis, Ecole des Mines de Paris, 1978.

[110] C. Lantuéjoul and F. Maisonneuve. Geodesic methods in quantitative image analysis. Pattern Recognition, 17:177–187, 1984.

[111] G. Lebrun, C. Charrier, O.Lezoray, C. Meurie, and H. Cardot. A fast and efficient segmentation scheme for cell microscopic image. In World Congress of Cellular and Molecular Biology, pages 35–36, 2005.

[112] F. Lefranc, Brotchi J., and Kiss R. Possible future issues in the treatment of glioblastomas: special emphasis on cell migration and the resistance of migrating glioblastoma cells to apoptosis. J. Clin. Oncol, 23(10):2411–22, 2005 Apr 1.

[113] O. Lezoray. Segmentation d'images par morphologie mathématique et classification de données par réseaux de neurones : Application à la classification de cellules en cytologie des séreuses. Ph.d thesis, University of Caen, 2000.

[114] O. Lezoray. Supervised automatic histogram clustering and watershed segmentation. application to microscopic medical images. Image Analysis and Stereology, 22:113–120, 2003.

[115] O. Lezoray and H. Cardot. Cooperation of color pixel classification schemes and color watershed: a study for microscopic images. IEEE Trans. Image Processing, 11(7):783–789, July 2002.

[116] K. Li, Miller E. D., and Weiss L. E. et al. Online tracking of migrating and proliferating cells in phase-contrast microscopy. In Proceedings of the 2006 IEEE Conference on Computer Vision and Pattern Recognition, pages 65–72, June 2006.

[117] P.-S. Liao, T.-S. Chen, and P.-C. Chung. A fast algorithm for multilevel thresholding. J. Inf. Sci. Eng, 17(5):713–727, 2001.

[118] N. Lichtenstein, B. Geiger B., and Kam Z. Quantitative analysis of cytoskeletal organization by digital fluorescent microscopy. Cytometry Part A, 54A(1):8–18, 2003.

[119] G. Lin, U. Adiga, K. Olson, J. F. Guzowski, C. A. Barnes, and B. Roysam. A hybrid 3D watershed algorithm incorporating gradient cues and object models for automatic segmentation of nuclei in confocal image stacks. Cytometry Part A, 56A(1):23–36, 2003.

[120] Z. Lin, J. S. Jin, and H. Talbot. Unseeded region growing for 3D image segmentation. In Selected papers from Pan-Sydney Area Workshop on Visual Information Processing (VIP2000), volume 2, pages 31–37, 2001.

[121] J. Lindblad, C. Wählby, E. Bengtsson, and A. Zaltsman. Image analysis for automatic segmentation of cytoplasms and classification of rac1 activation. Cyclometry, 57(1):22–33, jan 2004.

[122] J. Logan, Edwards K., and Saunders N. Real-Time PCR: Current Technology and Applications. Caister Academic Press, 2009.

[123] X. Long, W. L. Cleveland, and Y. L. Yao. A new preprocessing approach for cell recognition. IEEE Transactions on Information Technology in Biomedicine, pages 407–412, 2005.

[124] X. Long, Cleveland W. L., and Yao Y. L. Effective automatic recognition of cultured cells in bright field images using fisher's linear discriminant preprocessing. Image and Vision Computing, 23(13):1203–1213, nov 2005.

[125] David G. Lowe. Distinctive image features from scale-invariant keypoints. International Journal of Computer Vision, 60(2):91–110, 2004.

[126] Mareel M. and Leroy A. Clinical, cellular, and molecular aspects of cancer invasion. Physiol. Rev, 83(2):337–76, 2003 Apr.

[127] H. T. Madhloom, S. A. Kareem, H. Ariffin, A. A. Zaidan, H. O. Alanazi, and B. B. Zaidan. An automated white blood cell nucleus localization and segmentation using image arithmetic and automatic threshold. 2010.

[128] M. Malo, A. Cartier-Michaud, C. Fabre-Guillevin, G. Hutzler, F. Delaplace, G. Barlovatz-Meimon, and A. Lesne. When a collective outcome triggers a rare individual event: a mode of metastatic process in a cell population. Mathematical Population Studies, 2010.

[129] M. Malo, C. Charrière-Bertrand, E. Chettaoui, and G. Barlovatz-Meimon et al. The pai-1 swing : Microenvironment and cancer cell migration. C.R. Biol, 329(12):938–944, 2006.

[130] N. Malpica, C. Ortiz, J.J. Vaquero, A. Santos, I. Vallcorba, J.M. García-Sagredo, and F. del Pozo. Applying watershed algorithms to the segmentation of clustered nuclei. Cytometry, 28(4):289–297, 1997.

[131] N Malpica, C O De Solórzano, J J Vaquero, A Santos, I Vallcorba, J M García-Sagredo, and F Del Pozo. Applying watershed algorithms to the segmentation of clustered nuclei. Cytometry, 28(4):289–297, 1997.

[132] B. S. Manjunath and W. Y. Ma. Texture features for browsing and retrieval of image data. IEEE Trans. Pattern Analysis and Machine Intelligence, 18(8):837–842, August 1996.

[133] P. Maragos and R. W. Schafer. Morphological filters–part II: Their relations to median, order-statistic, and stack filters. IEEE Trans. Acoustics, Speech and Signal Processing, ASSP-35(8):1170, 1987.

[134] P. Maragos and R. D. Ziff. Threshold superposition in morphological image analysis systems. IEEE Trans. Pattern Anal. Machine Intelligence, 12:498–504, 1990.

[135] T. Markiewicz, Osowski S., Patera J., and Kozlowski W. Image processing for accurate cell recognition and count on histologic slides. Analytical and quantitative cytology and histology, 28(5):281–291, October 2006.

[136] D. R. Martin. An empirical approach to grouping and segmentation. In Ph.D., 2002.

[137] G. J. McLachlan. Discriminant analysis and statistical pattern recognition / Geoffry J. McLachlan. Wiley-Interscience, Hoboken, N.J., 2004.

[138] C. Meurie, G. Lebrun, O. Lezoray, and A. Elmoataz. A comparison of supervised pixel-based color image segmentation methods, application in cancerology. October 13-15 2003.

[139] F. Meyer. Topographic distance and watershed lines. Signal Processing, 38(1):113–125, July 1994.

[140] F. G. Meyer and S. Beucher. Morphological segmentation. Journal of Visual Communication and Image Representation, 1(1):21–46, 1990.

[141] Y. Meyer. Oscillating patterns in image processing and in some nonlinear evolution equations., March 2001.

[142] T. Mijatovic, Gailly P., Mathieu V., and Decaestecker C. et al. Neurotensin is a versatile modulator of in vitro human pancreatic ductal adenocarcinoma cell (pdac) migration. Cell. Oncol., 29:315–326, 2007.

[143] N. Mohanty, T. M. Rath, and A. Lea et al. Lecture notes in computer science. In nternational Conference on Image and Video Retrieval, pages 589–598, 2005.

[144] L. Najman and M. Schmitt. Watershed of a continuous function. Signal Processing, 38:99–112, 1994.

[145] L. Najman and M. Schmitt. Geodesic saliency of watershed contours and hierarchical segmentation. IEEE Transactions on Pattern Analysis and Machine Intelligence, 18(12):1163–1173, 1996.

[146] T. W. Nattkemper, T. Twellmann, W. Schubert, and H. Ritter. Human vs. machine: Evaluation of fluorescence micrographs. Computers in Biology and Medicine, 33(1):31–43, 2003.

[147] I. Nyström, G. Sanniti di Baja, and S. Svensson, editors. Perimeter and Area Estimations of Digitized Objects with Fuzzy Borders, volume 2886 of Lecture Notes in Computer Science, 2003.

[148] E. Obser, S. Lepert, and S. Lelandais. Image processing for glass industry. In Int. Conf. on Quality Control by Artificial Vision (QCAV'98), Nagoya, Japan, November 1998.

[149] William of Ockham. Quaestiones et decisiones in quattuor libros sententiarum petri lombardi (ed. ). Lugd., i, dist. 27, qu. 2, K, 1495.

[150] A. Onn and Fidler I.J. Metastatic potential of human neoplasms. In Vivo, 16(6):423–9, 2002 Nov-Dec.

[151] S. Osher, A. Solé, and L. Vese. Image decomposition and restoration using total variation minimization and the H −1 norm, 2003.

[152] N. Otsu. A threshold selection method from gray-level histograms. IEEE Transactions on Systems, Man, and Cybernetics, 9(1):62–66, 1979.

[153] N. R. Pal and S. K. Pal. A review on image segmentation techniques. Pattern Recognition, 26(9):1277–1294, 1993.

[154] D. P. Panda and A. Rosenfeld. Image segmentation by pixel classification in (gray level, edge value) space. IEEE Trans. Comput, 27:875–879, September 1978.

[155] T. Pavlidis. Structural Pattern Recognition, volume 1 of Springer Series in Electrophysics. Springer-Verlag, 1977.

[156] T. Pavlidis. Hierarchies in structural pattern recogniton. Proceedings of IEEE, 67:737–744, 1979.

[157] T. Pavlidis and Y. T. Liow. Integrating region growing and edge detection. IEEE Trans. Pattern Analysis and Machine Intelligence, 12(3):225–233, March 1990.

[158] T.X. Pedersen, C.J. Pennington, K. Almholt, I.J. Christensen, B.S. Nielsen, D.R. Edwards, J. Rømer, K. Dano, and M. Johnsen. Extracellular protease mrnas are predominantly expressed in the stromal areas of microdissected mouse breast carcinomas. Carcinogenesis, 26(7):1233–1240, 2005.

[159] H. Peng, X. Zhou, F. Li, X. Xia, and S. T. C. Wong. Integrating multi-scale blob/curvilinear detector techniques and multi-level sets for automated segmentation of stem cell images. In ISBI, pages 1362–1365. IEEE, 2009.

[160] P. Perona and J. Malik. Scale-space and edge detection using anisotropic diffusion. IEEE Transactions on Pattern Analysis and Machine Intelligence, PAMI-12(7):629–639, jul 1990.

[161] S. Philipp-Foliguet and L. Guigues. Multi-scale criteria for the evaluation of image segmentation algorithms. Journal of Multimedia, 3(5), 2008.

[162] R. Pinzón, G. Garavito, L. Arteaga Y.Hata, and J.D García. Development of an automatic counting system for blood smears. In Proc. of the Cong. of the Spanish Biomed. Engg. Society, pages 45–49, 2004.

[163] S. Popescu. A voxel-based lidar method for estimating crown base height for deciduous and pine trees. Remote Sensing of Environment, 112(3):767–781, 2008.

[164] R.T. Proffitt, J.V. Tran, and C.P. Reynolds. A fluorescence digital image microscopy system for quantitating relative cell numbers in tissue culture plates. Cytometry, 24:204–213, 1996.

[165] G. P. Qiu and F. J. Z. Feng. Compressing histogram representations for automatic colour photo categorization. Pattern Recognition, 37(11):2177–2193, November 2004.

[166] C. Rambabu and I. Chakrabarti. An efficient immersion-based watershed transform method and its prototype architecture. Journal of Systems Architecture, 53(4), 2007.

[167] P. Ranefall, K. Wester, and E. Bengtsson. Automatic quantification of immunohistochemically stained cell nuclei using unsupervised image analysis. Anal Cell Pathol, 16(1):29–43, 1998.

[168] M. Ranzato, P. E. Taylor, and J. M. House et al. Automatic recognition of biological particles in microscopic images. Pattern Recognition Letters, 28(1):31–39, January 2007.

[169] X. Ren and J. Malik. Learning a classification model for segmentation. In Proceedings of the Ninth IEEE International Conference on Computer Vision - Volume 2, ICCV '03, pages 10–18, 2003.

[170] B.D. Ripley. Pattern Recognition and Neural Networks. Cambridge University Press, Cambridge, 1996.

[171] Karsten Rodenacker and Ewert Bengtsson. A feature set for cytometry on digitized microscopic images. Anal Cell Pathol, 25(1):1–36, 2003.

[172] C. Rosello, P. Ballet, E. Planus, and Tracqui P. Model driven quantification of individual and collective cell migration. Acta Biotheor, 52(4):343–363, Apr 2004.

[173] A. Rosenfeld and J. Pfaltz. Distance functions on digital pictures. Pattern Recognition, 1:33–61, 1968.

[174] P. K. Sahoo, S. Soltani, A. K. C. Wong, and Chen. A survey of thresholding techniques. Comput. Vision Graph. Image Process, 41:233–260, February 1988.

[175] N. D. Salih, D. Chek, L. Ngo, and H. Mellah. 2d object description with discrete segments, 2006.

[176] F. Schnorrenberg, C. S. Pattichis, K. C. Kyriacou, and C. Schizas. Computer-aided detection of breast cancer nuclei. IEEE Transactions on Information Technology in Biomedicine, 1(2):128–140, 1997.

[177] J. Selinummi, J. Seppälä, O. Yli-Harja, and J. A. Puhakka. Software for quantification of labeled bacteria from digital microscope images by automated image analysis. BioTechniques, 39(6):859–863, 2005.

[178] S. Sergent-Tanguy, Chagneau C., Neveu I., and Naveilhan P. Fluorescent activated cell sorting (facs): a rapid and reliable method to estimate the number of neurons in a mixed population. Journal of Neuroscience Methods, pages 73–79, 2003.

[179] J. Serra. Image Analysis and Mathematical Morphology. 1982.

[180] J. Serra. Image analysis and mathematical morphology. Volume 2: Theoretical advances. Academic Press, London, 1988.

[181] J. Serra and L. Vincent. An overview of morphological filtering. Circuits Systems Signal Process, 11(1):47–108, 1992.

[182] M. Sezgin and B. Sankur. Survey over image thresholding techniques and quantitative performance evaluation. Journal of Electronic Imaging, 13(1):146–168, 2004.

[183] Linda G. Shapiro. Connected component labeling and adjacency graph construction. In T. Y. Kong and A. Rosenfeld, editors, Topological Algorithms for Digital Image Processing, volume 19 of Machine Intelligence and Pattern Recognition, pages 1 – 30. North-Holland, 1996.

[184] H. Sheikh, Zhu B., and Micheli-Tzanakou E. Blood cell identification using neural networks. pages 119–120, 1996.

[185] I. Simon, Pound C.R., Partin A.W., Clemens J.Q., and Christens-Barry W.A. Automated image analysis for detecting prostatic cell boundaries and quantifying cell motility. Cytometry, 31:287–294, 1998.

[186] I. Simon and Pound C. R. et al. Automated image analysis system for detecting boundaries of live prostate cancer cells. Cytometry, 31:287–294, 1998.

[187] S.W.S Sio and W. Sun et al. Malariacount: An image analysis-based program for the accurate determination of parasitemia. Journal of Microbiological Methods, 68(1):11–18, 2007.

[188] S.W.S Sio, Sun W., Kumar S., Bin W. Z., Tan S. S., Ong H., and et al. Malariacount: An image analysis-based program for the accurate determination of parasitemia. Journal of Microbiological Methods, 68(1):11–18, 2007.

[189] P. Soille. Constrained connectivity for hierarchical image decomposition and simplification. IEEE Trans. Pattern Analysis and Machine Intelligence, 30(7):1132–1145, July 2008.

[190] C. Ortiz De Solorzano, R. Malladi, S. A. Lelièvre, and S. J. Lockett. Segmentation of nuclei and cells using membrane related protein markers. Journal of Microscopy, 201(3):404–415, 2001.

[191] C. Spearman. The Abilities of Man: Their Nature and Measurement. Macmillan NY, 1927.

[192] J.-L. Starck, M. Elad, and D.L Donoho. Image decomposition: Separation of texture from piecewise smooth content, 2003.

[193] D. J. Stephens and Allan V. J. Light microscopy techniques for live cell imaging. Science, 300 (5616):82, Apr 4 2003.

[194] H. Sun, J. Yang, and M. Ren. A fast watershed algorithm based on chain code and application in image segmentation. Pattern Recognition, 26(69):1266–1274, 2005.

[195] S. O. Suzuki and Iwaki T. Dynamic analysis of glioma cells: looking into “movement phenotypes”. Neuropathology, 25(3):254–262, Sep 2005.

[196] J. R. Swedlow, I. Goldberg, E. Brauner, and P. K. Sorger. Informatics and quantitative analysis in biological imaging. Science, 300(5616):100–102, April 2007.

[197] J. Taylor, Hickson J., Lotan T., Yamada D.S., and Rinker-Schaeffer C. Using metastasis suppressor proteins to dissect interactions among cancer cells and their microenvironment. Cancer and Metastasis Reviews, 27(1):67–73, 2008 May.

[198] R Development Core Team. R: A language and environment for statistical com- puting. In R Foundation for Statistical Computing, Vienna, Austria, 2005. http://www.R-project.org. ISBN 3-900051-07-0.

[199] J. Theerapattanakul, Plodpai J., and Pintavirooj C. An efficient method for segmentation step of automated white blood cell classification. In Proceedings of the IEEE TENCON, pages 191–194, 2004.

[200] F. J. Theis, Z. Kohl, H. G. Kuhn, H. G. Stockmeier, and E. W. Lang. Automated counting of labelled cells in rodent brain section images. In Proc. of Int. Conf. Biomedical Engineering (BioMED), pages 209–212, 2004.

[201] F. J. Theis, Z. Kohl, H. G. Kuhn, H. G. Stockmeier, and E. W. Lang. Zane - an algorithm for counting labelled cells in section images. In Proc. of European Signal Processing Conference (EUSIPCO), pages 312–319, 2004.

[202] F.T. Theis, Kohlb Z., Guggenbergera C., Kuhnb G., and Lang E.W. Automated counting of newborn cells during neurogenesis. Bentham Science Publishers, 2009.

[203] J.P. Thiery. Epithelial-mesenchymal transitions in tumour progression. Nat Rev Cancer, 2(6):442–454, 2002.

[204] J.-P. Thiran and B. Macq. Morphological feature extraction for the classification of digital images of cancerous tissues. IEEE Transactions on Biomedical Engineering, 43(10):1011–1020, 1996.

[205] K. Tieu and P. Viola. Boosting image retrieval. International Journal of Computer Vision, 56(1-2):17–36, January 2004.

[206] A. Trémeau and P. Colantoni. Regions adjacency graph applied to color image segmentation. IEEE Transactions on Image Processing, 9:735–744, 2000.

[207] G. L. Turin. An introduction to matched filters. IRE Transactions on Information Theory, pages 310–329, 1960.

[208] Kindratenko V. V. On using functions to describe the shape. J. Math. Imaging and Vision, 18:225–245, 2003.

[209] L. A. Vese and S. J. Osher. Modeling textures with total variation minimization and oscillating patterns in image processing. J. Sci. Computing, 19:553–572, 2003.

[210] V. Vigneron, Lelandais S., Charriere-Bertrand C., Malo M., Ugon A., and Barlovatz-Meimon G. Pro or cons local vs. global imagery information for identifying cell migratory potential. In 15th EUSIPCO, pages 443–448, 2007.

[211] V. Vigneron, T.Q. Syed, G. Barlovatz-Meimon, M. Malo, C. Montagne, and S Lelandais. Adaptive filtering and hypothesis testing: Application to cancerous cells detection. Pattern Recognition Letters, 31(14):2214–2224, 2010.

[212] E. Vincan, T. Brabletz, M.C. Faux, and R.G. Ramsay. A human three-dimensional cell line model allows the study of dynamic and reversible epithelial-mesenchymal and mesenchymal-epithelial transition that underpins colorectal carcinogenesis. Cells Tissues Organs, 185:20–28, 2007.

[213] L. Vincent. Morphological grayscale reconstruction in image analysis: applications and efficient algorithms. IEEE Transactions on Image Processing, 2(2):176–201, April 1993.

[214] L. Vincent. Mathematical Morphology and its Applications to Image and Signal Processing. Kluwer Academic Publishers, 2000.

[215] L. Vincent and Soille P. Watersheds in digital spaces: an efficient algorithm based on immersion simulations. IEEE Trans. Patt. Anal. Mach. Intell, 13(6):583–598, 1991.

[216] L. Vincent and P. Soille. Watersheds in digital spaces: an efficient algorithm based on immersion simulations. IEEE Trans. Patt. Anal. Mach. Intell, 13(6):583–598, 1991.

[217] L. Vincet. Segmentation et Mise en Correspondance de R'egions de Paires d'Images Stéréoscopiques. PhD thesis, Université Paris IX - Dauphine, July 1991.

[218] C. Wahlby. Algorithms for Applied Digital Image Cytometry. PhD thesis, Uppsala University, 2003.

[219] C. Wählby, I.-M. Sintorn, F. Erlandsson, G. Borgefors, , and E. Bengtsson. Combining intensity, edge, and shape information for $2d$ and $3d$ segmentation of cell nuclei in tissue sections. Journal of Microscopy, 215(1):67–76, 2004.

[220] C. Wählby, I.-M. Sintorn, and et al F. Erlandsson. Combining intensity, edge, and shape information for 2*d* and 3*d* segmentation of cell nuclei on tissue sections. Journal of Microscopy, 215(1):67–76, July 2004.

[221] C. L. Weber. Elements of detection and signal design. Mc-Graw Hill Book Company, NY, 1968.

[222] P. D. Wendt, E. J. Coyle, and N. C. Gallagher Jr. Stack filters. IEEE Transactions on Acoustics, Speech, and Signal Processing, 34:898–911, 1986.

[223] C. Wählby, J. Lindblad, M. Vondrus, E. Bengtsson, and L. Björkesten. Algorithms for cytoplasm segmentation of fluorescence labelled cells. In Analytical Cellular Pathology, volume 24(2), pages 101–111, 2002.

[224] C.E. Wilkins-Port, C. E. Higgins, J. Freytag, S.P Higgins, J.A. Carlson, and P.J Higgins. Pai-1 is a critical upstream regulator of the tgf-1/egf-induced invasive phenotype in mutant p53 human cutaneous squamous cell carcinoma. Journal of Biomedicine and Biotechnology, 2007:1233–1240, 2007.

[225] D. L. Wilson, A. J. Baddeley, and R. A. Owens. A new metric for grey-scale image comparison. International Journal of Computer Vision, 24(1), 1997.

[226] I.P. Witz. Tumor-microenvironment interactions: dangerous liaisons. Adv Cancer Res, 100:203–229, 2008.

[227] Q. Xu and C. Karam. A distributed canny edge detector and its implementation on fpga. In Digital Signal Processing Workshop and IEEE Signal Processing Education Workshop, 2011 IEEE, pages 500–505, Jan 2011.

[228] F. Yang, M. A. Mackey, F. Ianzini, G. Gallardo, and M. Sonka. Cell segmentation, tracking, and mitosis detection using temporal context. Med Image Comput Comput Assist Interv, 8(1):302–309, 2005.

[229] Q. Yang and B. Parvin. CHEF: convex hull of elliptic features for 3D blob detection. In ICPR, pages II: 282–285, 2002.

[230] W. A. Yasnoff, W. Galbraith, and J. W. Bacus. Error measures for objective assessment of scene segmentation algorithms. AQC, 1:107–121, 1979.

[231] D. Yi, L. Smith, and M. Smith et al. A computer assisted diagnosis system for malignant melanoma using 3D skin surface texture features and artificial neural network. Int. J. of Modelling, 9:370–381, May 13 2010.

[232] M. Yi-De, D. Ro-Lan, L. Li, and Z. Zai-Fen. An counting and segmentation method of blood cell image with logical and morphological feature of cell. Chinese Journal of Electronics SCI&EI, 2001.

[233] L. Zadeh. Fuzzy sets. Information and Control, 8:338–353, 1965.

[234] M.H. Zaman, Trapani L.M., Sieminski A.L., Mackellar D., and et al. Migration of tumor cells in 3d matrices is governed by matrix stiffness along with cell-matrix adhesion and proteolysis. Biophys J, 103(29):10889–10894, 2006.

[235] F. Zernike. Diffraction theory of the cut procedure and its improved form, the phase contrast method. Physica, 1:689–704, 1934.

[236] F. Zernike. Phase-contrast, a new method for microscopic observation of transparent objects. Physica, 9:686–986, 1942.

[237] F. Zernike. How i discovered phase-contrast, 1953.

[238] H. Zhang, J. E. Fritts, and S. A. Goldman. Image segmentation evaluation: A survey of unsupervised methods. Computer Vision and Image Understanding, 110(2):260–280, May 2008.

[239] Y. J. Zhang. A survey on evaluation methods for image segmentation. Pattern Recognition, 29(8):1335–1346, August 1996.

[240] Y. J. Zhang. Evaluation and comparison of different segmentation algorithms. Pattern Recognition Letters, 18:963–974, 1997.

[241] Y. J. Zhang and J. J. Gerbrands. Objective and quantitative segmentation evaluation and comparison. Signal Processing, 39(1-2):43–54, 1994.

[242] Q. Zheng, Milthorpe B. K., and Jones A. S. Direct neural network application for automated cell recognition. Cytometry A, 57(1):1–9, January 2004.

[243] C. Zimmer, Labruyere E., and Yedid V. M. et al. Improving active contours for segmentation and tracking of motile cells in videomicroscopy. In ICPR, pages II: 286–289, 2002.

[244] K. H. Zou, S. K. Warfield, A. Bharatha, C. M. C. Tempany, M. R. Kaus, S. J. Haker, W. M. Wells, F. A. Jolesz, and R. Kikinis. Statistical validation of image segmentation quality based on a spatial overlap index. Academic Radiology, 11(2):178–189, 19200493.

# Annex A: Morphological image reconstruction

Reconstruction is a very useful operator provided by mathematical morphology [179, 180]. Although it can easily be defined in itself, it is often presented as part as a set of operators known as *geodesic* ones [110]. The reconstruction transformation is relatively well-known in the binary case, where it simply extracts the connected components of an image which are "marked" by another image (seeFig. 5.10). However, reconstruction can be defined for grayscale images as well. In this framework, it turns out to be particularly interesting for several filtering, segmentation and feature extraction tasks.

### Reconstruction for binary images

Let $I$ and $J$ be two binary images defined on the same discrete domain $D_I$ and such that $J \subseteq I$. In terms of mappings, this means that: $\forall p \in D; J(p) = 1 \Rightarrow I(p) = 1$. $J$ is called the marker image and $I$ is the mask. Let $I_1, I_2, ...I_n$ be the connected components of $I$.

**Definition** The reconstruction $I(J)$ of mask $I$ from marker $J$ is the union of the connected components of $I$which contain at least a pixel of $J$:

$$\rho_I(J) = \cup_{J \cap I_K \neq \emptyset} I_K \tag{5.16}$$

Fig. 5.10 illustrates this extremely simply but extremely useful transform.

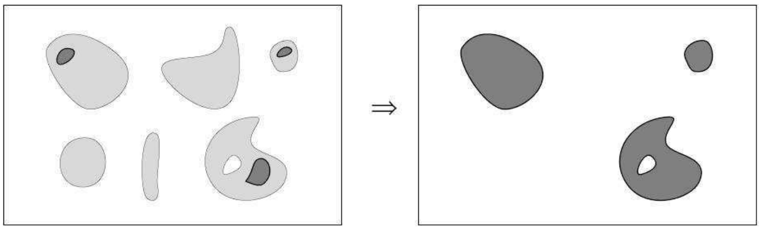


**Figure 5.10:** *Binary reconstruction from markers*

### Grayscale reconstruction

It is known that at least in the discrete case that any increasing transformation defined for binary images can be extended to grayscale images [179, 180, 181, 222]. By increasing, we mean a transformation $\psi$ such that $\forall X, Y \subset \mathbb{Z}^2, Y \subseteq X \Rightarrow \psi(Y) \subseteq \psi(X)$, In order to extend such a transformation $\psi$ to grayscale images $I$

taking their values in $0, 1, ..., N-1$, it suffices to consider the successive thresholds $T_k(I)$ of $I$, for $k = 0$ to $N-1$:

$$T_k(I) = \{p \in D_I | I(p) \geq k\}. \tag{5.17}$$

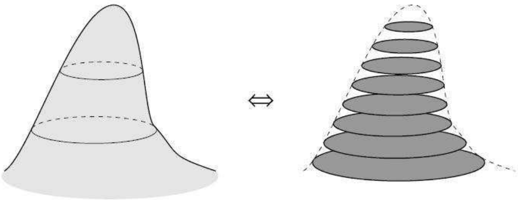


**Figure 5.11:** *Threshold decomposition of a greyscale image*

They are said to constitute the threshold decomposition of $I$ [133, 134]. As illustrated by Fig. 5.11, these sets obviously satisfy the following inclusion relationship:

$$\forall k \in [1..N-1], T_k(I) \subseteq T_{k-1}(I). \tag{5.18}$$

When applying the increasing operation $\psi$ to each of these sets, their inclusion relationships are preserved. Thus, we can now extend $\psi$ to grayscale images as follows:

$$\forall p \in D_I, \psi(I)(p) = max\{k \in [0, N-1] | p \in \psi(T_k(I))\}. \tag{5.19}$$

In the present case, binary geodesic reconstruction is an increasing transformation in that it satisfies:

$$Y_1 \subseteq Y_2, X_1 \subseteq X_2; Y_1 \subseteq X1, Y2 \subseteq X2 \Rightarrow \rho X_1(Y_1) \subseteq X_2(Y_2). \tag{5.20}$$

Therefore, following the threshold superposition principle of equation 5.17, we define grayscale reconstruction as follows [213]:

**Definition** (grayscale reconstruction): Let $J$ and $I$ be two grayscale images defined on the same domain, taking their values in the discrete set $\{0, 1, ..., N-1\}$ and such that $J \leq I(i.e., foreachpixelp \in D_I; J(p) \leq I(p))$. The grayscale reconstruction $\rho I(J)$ of $I$ from $J$ is given by:

$$\forall p \in D_I, \rho I(J)(p) = max\{k \in [0, N-1] | p \in \rho_{T_k(I)}(T_k(J))\}. \tag{5.21}$$

Fig. 5.12 illustrates this transformation. Just like binary reconstruction extracts those connected components of the mask which are marked, grayscale reconstruction extracts the peaks of the mask which are marked by the marker-image.

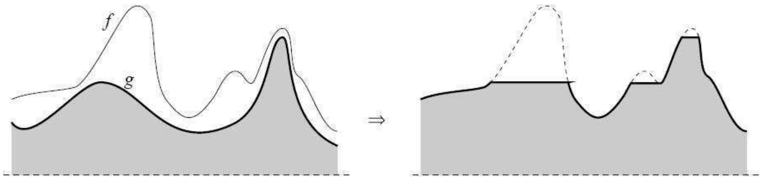


**Figure 5.12:** *Greyscale reconstruction of image f from marker g*

### Regional maxima and dome extraction

Reconstruction turns out to provide a very efficient method to extract regional maxima and minima from grayscale images. Furthermore, the technique extends to the determination of "maximal structures", which we call h-domes and h-basins. Let us first briefly review the notion of regional maximum:

**Definition (regional maximum)** A regional maximum $M$ of a grayscale image $I$ is a connected components of pixels with a given value $h$ (plateau at altitude $h$), such that every pixel in the neighborhood of $M$ has a strictly lower value.

Regional maxima should not be mistaken with local maxima. Recall that a pixel $p$ of $I$ is a local maximum for grid $G$ if and only if its value $I(p)$ is greater or equal to that of any of its neighbours. All the pixels belonging to a regional maximum are local maxima, but the converse is not true: for example, a pixel $p$ belonging to the inside of a plateau is a local maximum, but the plateau may have neighboring pixels of higher altitude and thus not be a regional maximum.

An alternative definition can also be proposed for the notion of regional maximum: Definition A regional maximum at altitude $h$ of grayscale image $I$ is a connected component $C$ of $T_h(I)$ such that $C \cap T_{h+1}(I) = \emptyset$. (Recall from eq. that $T_h(I)$ is threshold of I at level h.)

Determining the regional maxima of a greyscale image is relatively easy and several algorithms have been proposed in literature. One of the most efficient methods makes use of grayscale reconstruction and is based on the following proposition:

**Proposition 1.1** The (binary) image $M(I)$ of the regional maxima of $I$ is given by:

$$M(I) = I - \rho_I(I - 1). \tag{5.22}$$

Proof : According to the definition above, a connected component $C$ of $T_h(I)$ is a maximum at level $h$ if and only if $C \cap T_{h+1}(I) = C \cap T_h(I - 1) = \emptyset$. In other words, the set $M_h$ of the pixels belonging to a maximum of I at altitude h is given by:

$$M_h = T_h(I)\ \rho_{T_h(I)}(T_h(I - 1)). \tag{5.23}$$

Now, for any $h$, $h\prime$, $h \neq h\prime$, $M_h \cap M_{h\prime} = \emptyset$. This means that by replacing the set difference ( ) by an algebraic diffierence and using the threshold superposition principle, formula 5.16 can be extended to the grayscale case. This proposition is illustrated by Fig. 5.13.

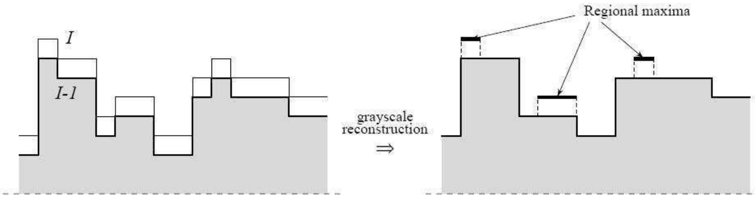


**Figure 5.13:** *Extracting the regional maxima of image I by its reconstruction from I-1*

Now, instead of subtracting value 1 in prop. 1.1 an arbitrary greylevel constant $h$ can be subtracted from $I$. This provides a useful technique for extracting "domes" of a given height, that we call h-domes. The following definition can be proposed:

Definition: The $h$-dome image $D_h(I)$ of the h-domes of a grayscale image $I$ is given by:

$$D_h(I) = I - \rho I(I - h). \tag{5.24}$$

Geometrically speaking, an h-dome can be interpreted the same way maxima are: an h-dome D of image $I$ is a connected component of pixels such that:

1. every pixel $p$ neighbor of $D$ satisfies: $I(p) < min\{I(q)|q \in D\}$,
2. $max\{I(q)|q \in D\} - min\{I(q)|q \in D\} < h$.

In addition, the value of pixel $p$ of $h - dome\ D$ in image $D_h(I)$ is equal to $I(p) - min\{I(q)|q \in D\}$. The $h$-dome transformation is illustrated on Fig. 5.14. Unlike classical top-hats, the h-dome transformation extracts light structures without involving any size or shape criterion. The only parameter (h) is related to the height of these structures. This characteristic is of interest for complex segmentation problems.

In the context of the hierarchic watershed transform we find it judicious to state a theorem about geodesic reconstruction which is implicitly used by all the authors.

**Theorem 5.0.1.** *Let $f$ and $g$ be two functions from $\mathbb{R}^n$ to $\mathbb{R}$, $f \geq g$.*

*Each regional minimum of the geodesic erosion $E^{\infty}_{f \bigwedge g}(g)$ contains at least one regional minumum of $f$.*

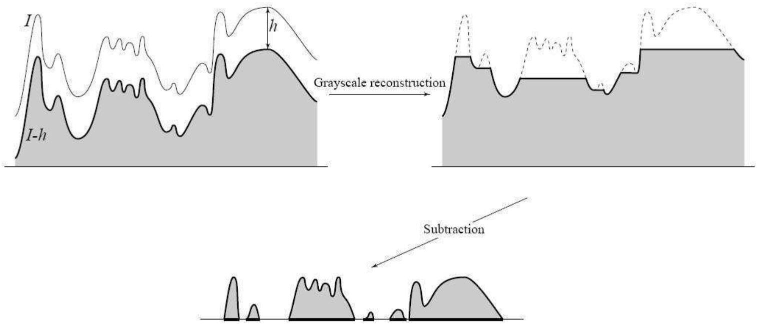


**Figure 5.14:** *Determining the h-domes of image I.*

That is to say, if $f \geq g$, the geodesic erosion $E^{\infty}_{f\bigwedge g}(g)$ can only surpass or merge regional minima of $f$. As the main problem in watershed segmentation is to suppress spurious minima, we understand why geodesic reconstruction is so crucial.

The various techniques of morphological image reconstruction form the foundation of several of the methods we have used in segmentation, such as in marking and swamping, and by extension it forms the crux of the cumulative hierarchy by multilevel marking algorithmic theory, as well as the determination of shaped-markers using $h$-domes.