Automatic Unstained Cell Detection in Bright Field Microscopy

Automatische Detektion ungefärbter Zellen in der Hellfeld-Mikroskopie

Der Technischen Fakultät
der Friedrich-Alexander-Universität
Erlangen-Nürnberg

zur

Erlangung des Doktorgrades Dr.-Ing.

vorgelegt von

Firas Mualla
aus
Latakia, Syrien

Als Dissertation genehmigt
von der Technischen Fakultät
der Friedrich-Alexander-Universität Erlangen-Nürnberg

Tag der mündlichen Prüfung:	28.11.2016
Vorsitzende des Promotionsorgans:	Prof. Dr.-Ing. Reinhard Lerch
Gutachter:	Prof. Dr.-Ing. Joachim Hornegger
	Prof. Dr. Benjamin Berkels

Abstract

Bright field microscopy is preferred over other microscopic imaging modalities whenever ease of implementation and minimization of expenditure are main concerns. This simplicity in hardware comes at the cost of image quality yielding images of low contrast. While staining can be employed to improve the contrast, it may complicate the experimental setup and cause undesired side effects on the cells. In this thesis, we tackle the problem of automatic cell detection in bright field images of unstained cells. The research was done in context of the interdisciplinary research project COSIR. COSIR aimed at developing a novel microscopic hardware having the following feature: the device can be placed in an incubator so that cells can be cultivated and observed in a controlled environment. In order to cope with design difficulties and manufacturing costs, the bright field technique was chosen for implementing the hardware. The contributions of this work are briefly outlined in the text which follows.

An automatic cell detection pipeline was developed based on supervised learning. It employs Scale Invariant Feature Transform (SIFT) keypoints, random forests, and agglomerative hierarchical clustering (AHC) in order to reliably detect cells. A *keypoint classifier* is first used to classify keypoints into *cell* and *background*. An intensity profile is extracted between each two nearby cell keypoints and a *profile classifier* is then utilized to classify the two keypoints whether they belong to the same cell (*inner* profile) or to different cells (*cross* profile). This two-classifiers approach was used in the literature. The proposed method, however, compares to the state-of-the-art as follows: 1) It yields high detection accuracy (at least 14% improvement compared to baseline bright field methods) in a fully-automatic manner with short runtime on the low-contrast bright field images. 2) Adaptation of standard features in literature from being pixel-based to adopting a keypoint-based extraction scheme: this scheme is sparse, scale-invariant, orientation-invariant, and feature parameters can be tailored in a meaningful way based on a relevant keypoint scale and orientation. 3) The pipeline is highly invariant with respect to illumination artifacts, noise, scale and orientation changes. 4) The probabilistic output of the profile classifier is used as input for an AHC step which improves detection accuracy. A novel linkage method was proposed which incorporates the information of SIFT keypoints into the linkage. This method was proved to be combinatorial, and thus, it can be computed efficiently in a recursive manner.

Due to the substantial difference in contrast and visual appearance between suspended and adherent cells, the above-mentioned pipeline attains higher accuracy in *separate learning* of suspended and adherent cells compared to *joint learning*. Separate learning refers to the situation when training and testing are done either only on suspended cells or only on adherent cells. On the other hand, joint learning refers to training the algorithm to detect cells in images which contain both suspended and adherent cells. Since these two types of cells coexist in cell cultures with shades of gray between the two terminal cases, it is of practical importance to improve joint learning accuracy. We showed that this can be achieved using two types of phase-based features: 1) physical light phase obtained by solving the transport of intensity equation, 2) monogenic local phase obtained from a low-passed axial derivative image.

In addition to the supervised cell detection discussed so far, a cell detection approach based on unsupervised learning was proposed. Technically speaking, supervised learning was utilized in this approach as well. However, instead of training the profile classifier using manually-labeled ground truth, a self-labeling algorithm was proposed with which ground truth labels can be automatically generated from cells and keypoints in the input image itself. The algorithm learns from *extreme* cases and applies the learned model on the *intermediate* ones. SIFT keypoints were successfully employed for unsupervised structure-of-interest measurements in cell images such as mean structure size and dominant curvature direction. Based on these measurements, it was possible to define the notion of extreme cases in a way which is independent from image resolution and cell type.

Kurzübersicht

Hellfeldmikroskopie wird immer dann anderen Mikroskopieverfahren vorgezogen, wenn großer Wert auf die Minimierung der Anschaffungskosten und die Einfachheit der Umsetzung gelegt wird. Diese Einfachheit der Hardware vermindert jedoch die Bildqualität und führt zu einem verringerten Kontrast in den erzeugten Bildern. Eine Einfärbung der Zellen kann zur Erhöhung des Kontrasts verwendet werden. Allerdings macht sie die Versuchsanordnung komplizierter und verursacht Nebenwirkungen auf die Zellen. In dieser Dissertation wurde das Problem der automatischen Detektion ungefärbter Zellen in Hellfeldmikroskopie-Bildern untersucht. Die Forschung fand im Rahmen des interdisziplinären Projekts COSIR statt. Ziel des Projekts COSIR war es, eine Mikroskop-Hardware zu entwickeln, mit der Zellkulturen innerhalb des Inkubators beobachtet werden können. Um Konstruktionschwierigkeiten zu vermeiden und Herstellungskosten gering zu halten, wurde die Hellfeldmikroskopie zur Umsetzung des COSIR-Projekts ausgewählt. Die Beiträge dieser Doktorarbeit sind im Folgenden zusammengefasst.

Basierend auf überwachtem Lernen wurde eine Pipeline zur automatischen Zelldetektion entwickelt. Sie verwendet Scale Invariant Feature Transform (SIFT), Random Forests, und die agglomerative hierarchische Clusteranalyse (AHC), um Zellen zuverlässig zu detektieren. Als erster Schritt wurde ein Keypoint-Klassifikator zur Unterscheidung zwischen Zell- und Hintergrund-Keypoints eingesetzt. Danach wurde ein Intensitätsprofil zwischen je zwei nebeneinanderliegenden Zell-Keypoints extrahiert. Ein Profil-Klassifikator wurde danach verwendet, damit die Profile entweder als *inner* (in derselben Zelle) oder *cross* (zwischen zwei Zellen) klassifiziert werden. Dieser Zwei-Klassifikatoren-Ansatz wurde bereits in der Literatur verwendet. Im Gegensatz zu anderen State-of-the-Art Algorithmen trägt der vorgeschlagene Ansatz das Folgende bei: 1) Die Zelldetektion ist vollautomatisch, arbeitet mit hoher Genauigkeit (mindestens 14% besser als Baseline Hellfeld-Ansätze) und in kurzer Zeit auf kontrastarmen Hellfeldbildern. 2) Pixelbasierte Standardmerkmale aus der Literatur wurden basierend auf SIFT-Keypoints angepasst. Dieser Ansatz ist dünnbesetzt, skaleninvariant, rotationsinvariant, und die Parameter der Merkmale können basierend auf der relevanten Vergrößerung und der relevanten Orientierung sinnvoll angepasst werden. 3) Die vorgeschlagene Pipeline ist weitgehend invariant gegenüber Beleuchtungsartefakten, Rauschen, und Änderungen der Vergrößerung oder der Orientierung. 4) Die probabilistische Ausgabe des Profil-Klassifikators wird als Eingabe eines AHC-Verfahrens genutzt, was die Genauigkeit der Detektion verbessert. Ein neues Linkage-Verfahren wurde dargestellt, das die Informationen der SIFT-Keypoints ins Linkage-Verfahren einbezieht. Es wurde bewiesen, dass dieses Verfahren kombinatorisch ist. Daher kann es effizient in rekursiver Weise berechnet werden.

Wegen des erheblichen Unterschieds zwischen adhärenten Zellen und Suspensionszellen sowohl im Kontrast als auch im Erscheinungsbild, liefert die oben aufgeführte Pipeline eine niedrigere Detektionsgenauigkeit bei gemeinsamem Lernen im Vergleich zum separaten Lernen. Separates Lernen bezieht sich auf die Situation, in der Training und Testen entweder nur auf adhärente Zellen oder nur auf Suspensionszellen angewandt werden. Auf der anderen Seite bezieht sich das gemeinsame Lernen auf die Situation, in der adhärente Zellen und Suspensionszellen zusammen in den Trainingsbildern enthalten sind. Da diese zwei Zelltypen in Zellkulturen koexistieren,

ist die Verbesserung des gemeinsamen Lernens wichtig für die Praxis. Wir haben gezeigt, dass dieses Ziel mit zwei Typen phasenbasierter Merkmale erreicht werden kann: 1) Die Phase des physikalischen Lichts, die man durch das Lösen der *Transport of Intensity Equation* erhält. 2) Die monogene lokale Phase, die basierend auf einer tiefpassgefilterten axialen Ableitung berechnet werden kann.

Zusätzlich zur bisher diskutierten überwachten Zelldetektion, wurde ein Ansatz zur unüberwachten Zelldetektion vorgeschlagen. Technisch gesehen, wurde auch hier überwachtes Lernen benutzt. Statt des Trainings des Profil-Klassifikators mit manuell gelabelten Ground-Truth-Daten, wurde ein Self-Labeling Algorithmus vorgeschlagen, mit dem Labels basierend auf Zellen und Keypoints im Bild automatisch erzeugt werden können. Der Algorithmus lernt aus extremen Fällen und wendet das gelernte Model auf die dazwischenliegenden Fälle an. SIFT-Keypoints wurden erfolgreich für Ermittlung der relevanten Strukturen (z. B. die mittlere Strukturgröße und die dominante Krümmungsrichtung) eingesetzt. Anhand dieser ermittelten Werte war es möglich, ein Konzept für die extremen Fälle zu definieren, das unabhängig von dem Zelltyp oder der Bildauflösung ist.

Acknowledgment

I would like to sincerely thank my supervisor Prof. Dr.-Ing. Joachim Hornegger for his support, encouragement, and critique. What I learned from him, especially from the way in which he approaches science, was immensely influential for this dissertation. I want, however, to emphasize that it was even more influential for the development of my personal scientific awareness. For that, I owe him a lifelong feeling of deepest gratitude.

Sincere thanks go to Prof. Dr.-Ing. Andreas Maier for reviewing my papers and spreading motivation in the group. I was impressed how he gets things done with a touch of creative simplicity. Very special thanks go to Simon Schöll for the great time we spent together in scientific and non-scientific discussions. The COSIR project would have failed without his infinite commitment and positive attitude. I am grateful to Prof. Dr. Benjamin Berkels for reviewing the dissertation in details and in a short time. I want also to thank Dr.-Ing. Stefan Steidl for his useful comments regarding the writing style, Dr. Elli Angelopoulou for proofreading one of my articles, and David Bernecker for the help in the TikZ graphics. I am grateful to Wilhelm Haas, my roommate at the LME, for the nice friendly atmosphere.

From outside the LME, I wish to thank Prof. Dr. Rainer Buchholz and Dr. Björn Sommerfeldt from the Institute of Bioprocess Engineering in Erlangen for preparing the cell cultures and helping us in dealing with microscopes. Sincere thanks go also to Dr. Jiyan Pan from the CMU for the fruitful discussions, Carlos Arteta from Oxford for phase contrast datasets, and Gabriele Becattini from the Italian Institute of Technology for the advice about his cell detection software.

Last but not least, I would like to thank my parents, sister, brother, and close friends for their support.

Firas

Contents

Chapter 1

Introduction

Automatic image-based cell detection approaches are indispensable in biomedical image analysis. They can be used for cell number estimation [Sjos 99, Louk 03], cell tracking [Li 08], initializing cell segmentation methods [Ali 12], and for extracting features which can be employed for other application-dependent tasks such as cell viability determination [Long 06, Van 13].

From an application point of view, the information obtained by cell detection approaches can be utilized in different medical, biological, and pharmaceutical fields including virology, toxicity tests, vaccine production, cancer research, and gene therapy. The role of automatic cell detection in these fields can be clarified in simple non-technical words: when cells are in a healthy state, they proliferate, otherwise they undergo a degradation in activity which may lead to cellular death. Therefore, by tracking the cell number after injecting a new factor (e. g. a chemical, a toxin, or a vaccine), the effect of this factor on the cells under study can be revealed. In addition to the cell number, the cell shape and the visual appearance are also important indicators of the cellular activity.

1.1 Microscopic Imaging and Research Context

There are many microscopic modalities which enable us to visualize cells. For instance, a bright field microscope utilizes cell absorption of light in order to form an image. A phase contrast microscope, on the other hand, visualizes the light phase-shift introduced by a specimen due to a difference in refractive index between the specimen and its surrounding medium. In fluorescence microscopy, fluorescent dyes are employed to improve the contrast by imaging the light which they *emit* upon being *excited* with a specific wavelength. Since microscopic modalities differ in the information they provide and the artifacts and/or limitations from which they suffer, each of them poses different challenges for automatic image analysis. For instance, in fluorescence microscopy, staining reshapes the cell detection problem as a relatively easy task due to high contrast in the acquired images. However, in some biological applications, it is desired to avoid staining for the following reasons: Firstly, it may induce side effects on cells [Lule 09]. Secondly, using fluorescence microscopy alone may lead to incomplete shape information. In fact, what we see using such a technique is the activity of fluorescent dyes which, in general, does not reveal structural

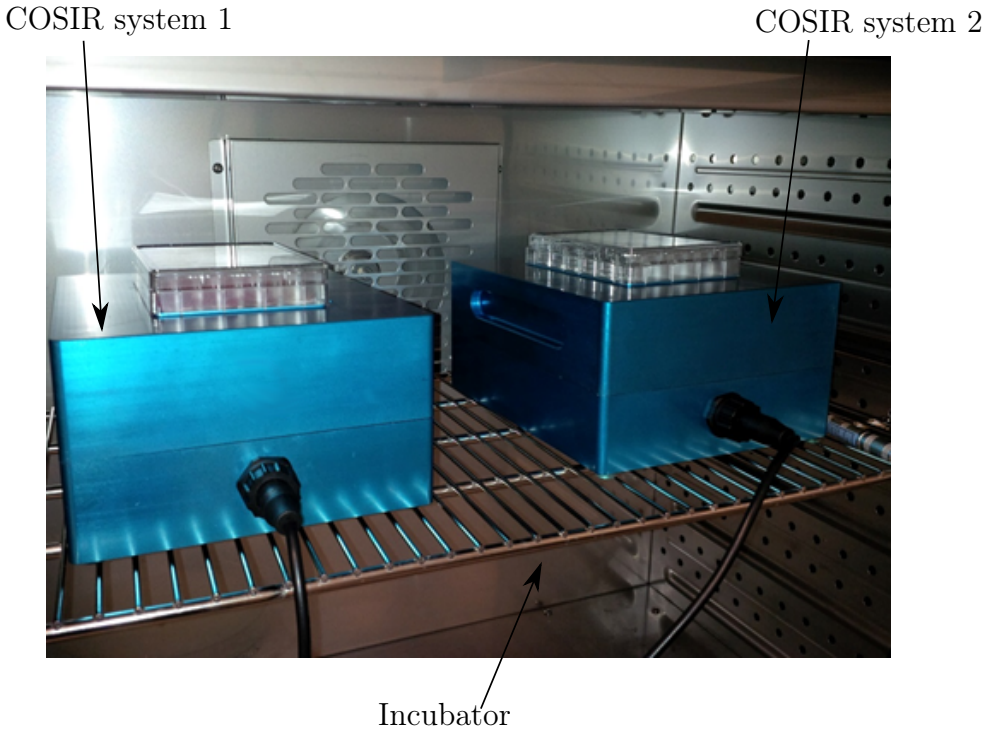


Figure 1.1: Two COSIR systems placed inside an incubator

information. Moreover, these fluorescent dyes do not always cover the entire cell, and it is thus not suited for cell boundary segmentation [Ali 07]. Without staining, cell detection is more challenging and sometimes very difficult [Opst 94, Long 05]. Microscope modalities will be thoroughly discussed in Chapter 2.

Typically, cells are cultivated within a special liquid called *culture medium*. Moreover, since cells require proper physical conditions to survive, they are placed inside a device known as *incubator*. This device enables biologists to control the physical conditions of the environment in which cells are cultivated such as temperature, CO_2 level, O_2 level, humidity, and other factors. Cell growth, i. e. the growth of cell population, in a cell culture depends on the surrounding physical conditions and the composition of the culture medium. Usually, in order to observe a cell culture by a microscope, a biologist needs to take the cell culture outside the incubator which causes undesired effects.

Most of the research in this thesis was performed in context of the interdisciplinary research project **COSIR**: Combination of Chemical-Optical Sensors and Image Recognition. In this project, a novel microscopic hardware was developed over a period of three years. The COSIR system (cf. Figure 1.1) contains 24 channels, each of which delivers an image of a single well of a microtiter plate (cf. Figure 1.2). A main attraction of COSIR is that it is designed to work inside an incubator, and hence, cell cultures can be observed in a controlled environment. The COSIR system is connected to a computer through which the acquired images can be observed using a special software developed by ASTRUM IT GmbH[1]. The hardware itself was

[1]ASTRUM IT GmbH, Erlangen: `http://www.astrum-it.de/`

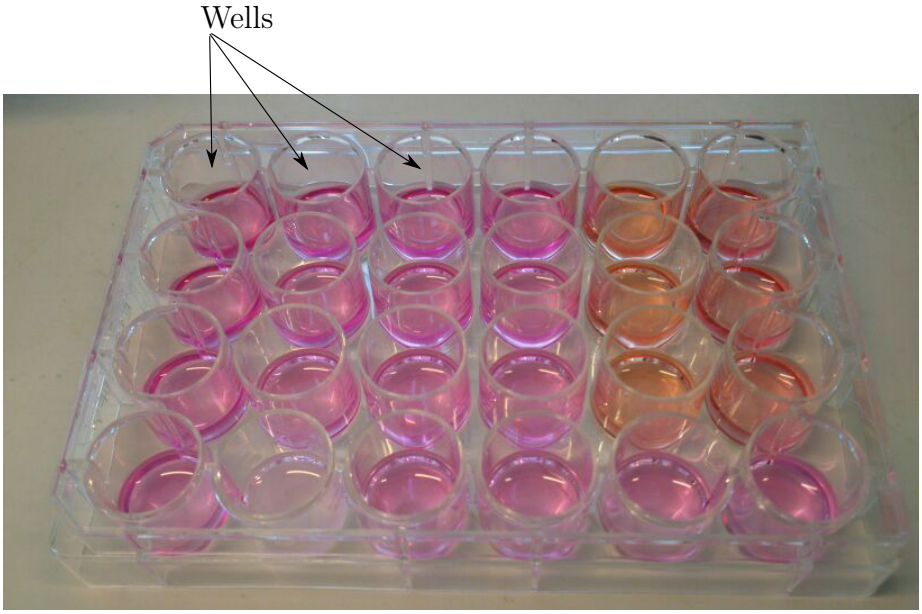


Figure 1.2: A 24-well microtiter plate containing a cell culture inside each well

developed by PreSens Precision Sensing GmbH[2] which was supported during the development by the Pattern Recognition Lab[3], ASTRUM IT GmbH, and the Institute of Bioprocess Engineering[4].

Due to the requirement that COSIR will operate inside an incubator, the hardware developers had to cope with real design difficulties. For instance, the system has to be limited in size so that it easily fits in an incubator. The size limitation and other factors such as manufacturing costs led to decisions which favor simple microscope components over complicated and costly ones. With this in mind, a bright field microscopic setup was adopted for COSIR as it is cheaper and easier to implement compared to the other microscope modalities. For the development and evaluation of our algorithms, we employed two types of bright field images: 1) *standard* bright field images obtained by Nikon Eclipse microscope (cf. Figure 1.3), 2) COSIR images obtained by COSIR prototype. The use of standard images was necessary for two reasons: 1) COSIR's image quality was instable during hardware development. 2) In order to guarantee that our algorithms can be used by anybody who has a standard bright field microscope.

Even though image analysis in bright field microscopy was the main focus of this thesis, many algorithmic concepts are general and can be applied to other modalities as well. For instance, both of our supervised [Mual 13a] and unsupervised [Mual 14b] cell detection approaches were successfully applied to phase contrast images.

Figure 1.4a shows a COSIR image at focus, while figures 1.4b and 1.4c show a *positively defocused* image and a *negatively defocused* image, respectively. An image is considered positively defocused when the microscope's objective approaches the object and negative otherwise [Ager 03]. Further details about this concept can be

[2]PreSens Precision Sensing GmbH, Regensburg: `http://www.presens.de`
[3]Pattern Recognition Lab, Erlangen: `http://www5.cs.fau.de/`
[4]Institute of Bioprocess Engineering, Erlangen: `http://www.bvt.cbi.uni-erlangen.de/`

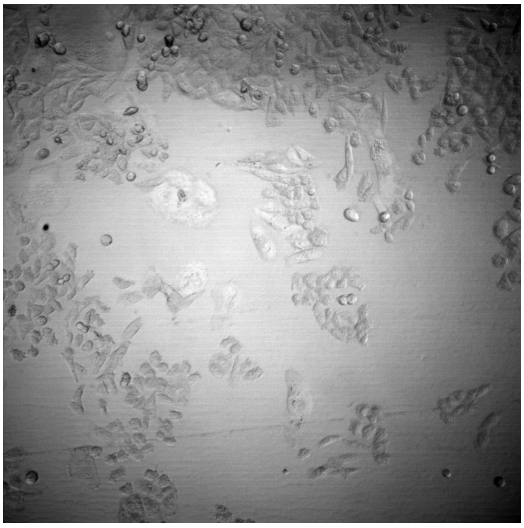
Figure 1.3: A standard Nikon Eclipse light microscope at the Institute of Bioprocess Engineering, Erlangen

found in Chapter 2. In this thesis, defocused images were employed for cell detection while both focused and defocused images were used for phase retrieval.

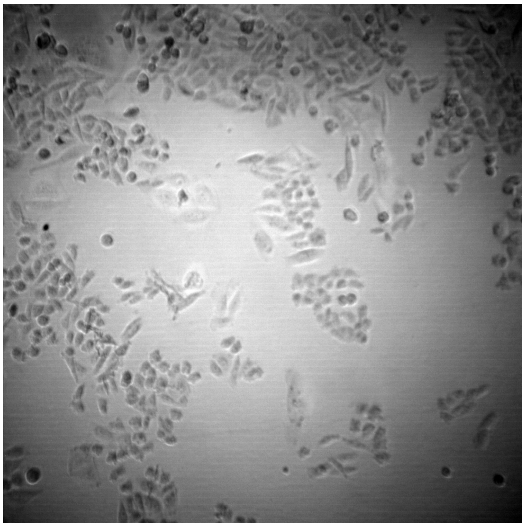
Figure 1.5 shows a standard bright field image at focus obtained by the microscope shown in Figure 1.3. The image contains both *adherent* and *suspended* cells. These two cell descriptors are used in cell biology to differentiate cells which adhere at the bottom of some surface (e. g. at the bottom of a microtiter-plate's well) and cells which freely float in the culture medium. Adherent cells, being thin due to their adherence at the surface, absorb much less light compared to cells in suspension. Therefore, in a bright field image, adherent cells exhibit low contrast compared to suspended cells.

1.2 State of the Art

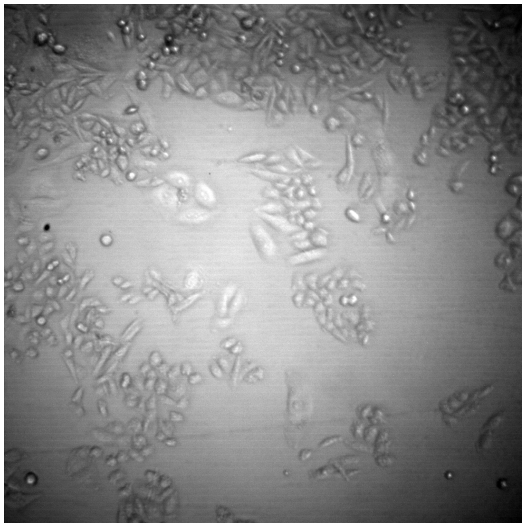
Unstained cell recognition in bright field images was frequently described in literature as a challenging problem [Opst 94, Long 05, Long 06, Tsch 08, Zari 11]. In general, cells exhibit a great diversity in shape and size. In addition, a simple microscopy technique such as bright field does not always offer sufficient contrast between cells and background [Tsch 08, Curl 04] (cf. Figure1.5). In fact, adherent cells are almost invisible at focus [Ager 03, Beca 11, Ali 07]. The contrast can be improved by defocusing the microscope (cf. Figure 1.4). This improvement in contrast can be interpreted by the



(a) A COSIR image at focus



(b) A positively defocused COSIR image



(c) A negatively defocused COSIR image

Figure 1.4: COSIR images of a CHO cell culture acquired at different focus levels. Contrast was linearly stretched for clarity.

so-called transport of intensity equation (TIE). In Section 2.6, the TIE and its related contrast will be explained in detail. Moreover, in Chapter 6, we will show how the TIE can be employed for improving joint learning of adherent and suspended cells.

In [Beca 11], a positively defocused image was segmented by applying the watershed algorithm on the distance transform of a thresholded image. An in-focus image was processed with the self-complementary top-hat [Soil 13] followed by thresholding and watershed. A negatively defocused image was preprocessed with a Canny edge detector before being analyzed by anisotropic contour completion [Gil 03]. These three results were then combined in order to select micro-injection points inside the cells. In [Curl 04], the three aforementioned images (positive, focused, and negative) where used to solve the TIE and obtain a phase map. Typically, the resulting phase map shows more contrast than the original images even though it may suffer from a

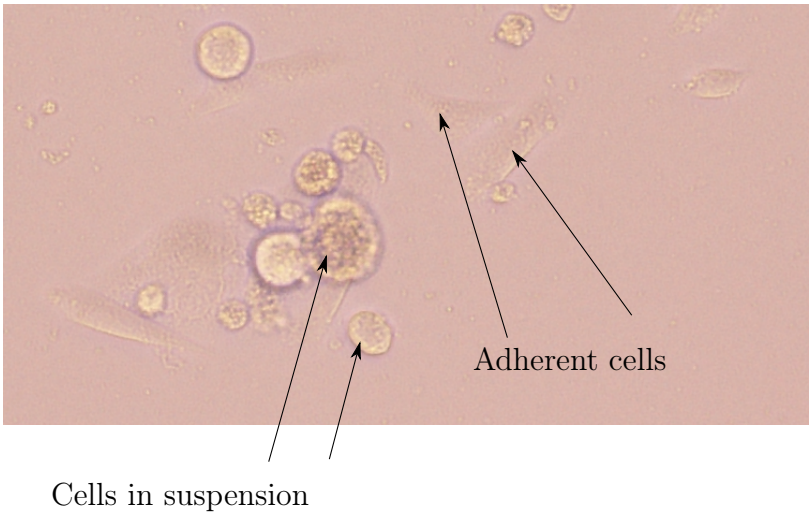


Figure 1.5: An image acquired with a standard bright field microscope showing the difference between adherent and suspended cells.

low-frequency bias field. Thresholding was then applied on the phase map in order to segment cells.

A considerable contribution to bright field image analysis of adherent ultra-thin cells was made by Rehan Ali and his colleagues in [Ali 10]. In [Ali 10], a link between the physical phase-shift of light and the local phase of a low-passed axial-derivative image was established. The physical phase-shift of light can be obtained by solving the TIE. On the other hand, the axial-derivative image can be estimated by subtracting two images at two different focus levels (e. g. subtracting the image shown in Figure 1.4b from the image shown in Figure 1.4c). The local phase is then obtained using the so-called monogenic signal framework with this axial derivative as input. A main feature of the utilized monogenic signal framework is its use of low-pass filters rather than band-pass filters which are typically employed for local phase estimation. This link between the two quantities, i. e. physical phase and local phase, is interesting as they express different concepts. The physical phase-shift of light is introduced when light passes through transparent objects such as ultra-thin adherent cells. It is related to the object thickness and the difference of refractive index between the object and its surrounding medium. On the other hand, local phase can be, informally speaking, perceived in the sense of phase in the short time Fourier transform. Essentially, it reveals local symmetry/asymmetry features. In Chapter 6, this connection between physical phase and local phase is profoundly explained.

In [Ali 12], in order to detect and segment cells, a positively defocused image was subtracted from a negatively defocused one. The difference image was thresholded and the result was post-processed with an appropriately chosen size filter implemented using morphological opening. Each connected component of the resulting mask was used to initialize a level-set evolution. The level-set driving force was based on monogenic signal features: local phase, local energy, and local orientation (cf. Chapter 6).

The aforementioned cell detection methods are based on image processing techniques. In contrast, there is a family of methods for cell detection that are learning-based. In these approaches, a classifier labels each pixel as either a cell or a non-cell

pixel. The output of the pixelwise classifier delivers a confidence map. The local maxima of the latter correspond to cell centers.

For training the classifier, features are extracted from a fixed-size patch sampled in the neighborhood of the pixel under investigation. Several papers share this common strategy despite differences in classifier model and image modality. For example, [Natt 99] used principal component analysis (PCA) features extracted from 15×15 sized patches which are then analyzed by an artificial neural network. In [Long 05], Fisher discriminant analysis was used instead of PCA. In [Long 06], a support vector machine (SVM) replaced the neural network.

An interesting pixelwise cell/background classification on phase contrast and differential interference contrast microscopy was suggested in [Yin 10]. In this approach, training was made more efficient by employing a clustering step on ground-truth feature vectors before using them to train a bag of classifiers. Fixed-size pixel patches are randomly extracted from training images and clustered into $\Bbbk$ categories, where $\Bbbk$ is a parameter of the algorithm. A patch is represented by a feature vector computed based on a local intensity histogram of the patch. Each of the resulting clusters is then used as ground truth to train a pixelwise cell/background classifier. This has the advantage of pushing training algorithms to learn how to discriminate feature vectors which are nearby in feature space. The training yields $\Bbbk$ classifiers, each of them is an expert in a specific visual appearance pattern. For classifying a patch from a test image, a feature vector is extracted and its distance[5] to each cluster is computed. Each classifier's output is then given a weight inversely proportional to the distance of the considered feature vector to the cluster on which this classifier was trained. The final decision for the considered patch is a weighted sum of the individual classifier decisions. Compared to AdaBoost [Freu 97] weights which are dependent on global classifier accuracies and are constant after training, this method incorporates information from test feature vectors into the process of aggregating individual classifiers. This property is partially similar to some boosting approaches in the machine learning literature which extend the scheme of AdaBoost such as WeightBoost [Jin 03] and iBoost [Kwek 02].

Most of these learning-based approaches require parameter tuning so that they can be successfully employed for cell detection. These parameters are related to thresholding the confidence map (the pixelwise classification result), applying morphological operators on the thresholding result, and/or searching for the maxima of the confidence map. In all learning-based approaches mentioned so far, the optimal size of the patch should conform to the mean cell size which is not always known a priori. In addition, the square neighborhood used in these methods, does not fit non-circular cells.

In the work of Jiyan Pan and his colleagues [Pan 09] on phase contrast microscopy, a major contribution to *point-based* cell detection was presented. In this work, maxima of convolution results with a bank of Laplacian filters are detected. The detected points with fluctuation energy below a specific threshold $thresh_1$ are discarded in order to reduce the number of points. Typically, these maxima are not well-localized inside cells. Therefore, a mean-shift algorithm is applied in order to refine their loca-

[5]Since the feature vectors are histograms, a similarity measure on probability distributions is employed, for instance, the Bhattacharyya coefficient.

tions. Mean-shift merges the points which are very close to each other. Additionally, it pushes the points which are close to cell boundaries toward cell centers. This is achieved by making use of the fact that cell centers tend to be darker than cell boundaries in phase contrast microscopy. Weights of the mean-shift kernel were thus made proportional to the darkness level. An SVM classifier is utilized to classify the points resulting from mean shift as either *cell points* or *background points*. Afterwards, another SVM classifier is employed to label each two nearby cell points as belonging to the same cell or to two different cells. A sigmoid function is fit on the SVM output (similar to Platt scheme in [Plat 99]) in order to convert distances to the SVM decision boundary to probability measures. Two points are considered to belong to the same cell if the probability obtained by the resulting Platt scheme is above a threshold $thresh_2$. This approach performs well in phase contrast microscopy, but it is sensitive to the critical thresholds $thresh_1$ and $thresh_2$. In [Pan 10], the method in [Pan 09] was extended and the dependency on thresholds was eliminated at the cost of some extra computation time. In the this approach, the two classification steps were performed jointly in a conditional random field (CRF) framework. The two aforementioned approaches require ground truth of segmented cells. In other words, cell borders should be delineated and each cell should have a distinguishing identifier in the ground-truth mask. In [Arte 12], maximally stable extremal regions (MSER) keypoints were utilized instead of the Laplacian maxima and a structured SVM [6] was used to learn a bijective mapping between the MSER regions and the ground-truth cell centers. Compared to [Pan 09] and [Pan 10], this approach has the advantage that it is easier to train because only cell centers are required as ground truth.

In some special cases, cellular images expose distinct characteristics which make it possible to adopt simplifying assumptions. For instance, images of vocal folds' epithelium or corneal endothelium exhibit two important properties (cf. Figure 1.6): Firstly, due to the nature of the imaged tissue, cells cover the whole scene. Therefore, the cell/background separation required in all approaches mentioned so far, is not necessary. Secondly, cells in a single image show a repetitive pattern. Consequently, in the Fourier transform of image intensity, this repetition will manifest itself as a peak at the fundamental frequency of the pattern. More specifically, in the spatial domain, $\cos(u_0 x + v_0 y)$, where x, y are the spatial dimensions, represents a sinusoid with angular frequency $\omega_0 = \sqrt{u_0^2 + v_0^2}$ along a direction defined by (u_0, v_0). In the Fourier domain, this corresponds to a peak at $\pm(u_0, v_0)$. If we assume isotropy, i. e. ω_0 is almost the same in all directions, these peaks will form a circular ring. This fact was exploited in [Fora 02] and [Rugg 05] for estimating cell density of donor corneas. The radius of the aforementioned ring is a measure of cell density, which is in turn a measure of the cornea quality. In [Mual 13b], this principle was applied on epithelial cell images of the vocal folds. However, instead of estimating the density, cells were detected by simply localizing minima in a band-pass filtered image. The pass-band was defined in terms of the Fourier-space ring. In [Bier 15], this method was extended by approaching the band-pass filter design as ring segmentation in the Fourier domain. Images of cell cultures in a medium, such as the images acquired for this thesis, do

[6] A structured SVM [Tsoc 04] is a generalization of the SVM model, in which labels may have arbitrary structures, e. g. sequences or trees.

not show a repetitive pattern. Therefore, the aforementioned approaches which are based on this assumption cannot be applied.

1.3 Contributions to the Progress of Research

The contributions of this thesis can be summarized as follows:

- A point-based supervised cell detection algorithm [Mual 13a] on bright field microscopy which utilizes the scale invariant feature transform (SIFT), two random forest classifiers, and an agglomerative hierarchical clustering step with a customized linkage method (cf. Section 3.3 and Section 5.5.1) in order to robustly detect cells in low-contrast bright field images. Compared to the state-of-the-art, it presents the following contributions:
  - The algorithm is fully automatic, i. e. no parameter tuning is required, neither in training nor in testing.
  - It achieves high detection rates (at least 14% improvement compared to baseline methods) in short runtimes on the low-contrast bright field images.
  - It is very robust against illumination artifacts. For instance, when the algorithm was tested on images perturbed with an illumination field whose energy is 100 times larger than the energy of the training image (which is

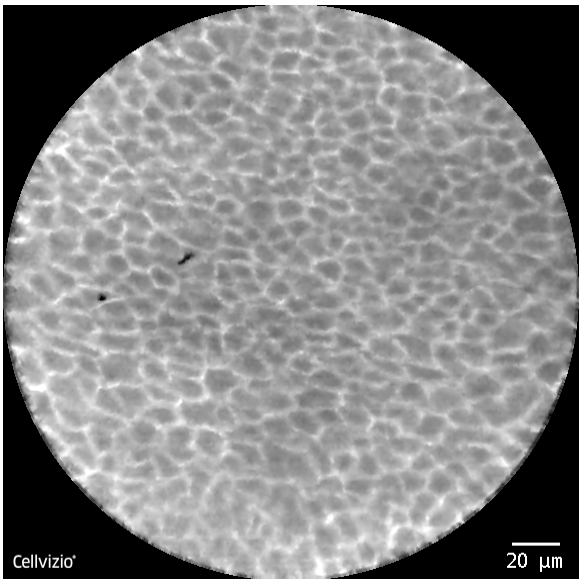


Figure 1.6: An endomicroscopy image of the vocal folds' epithelium acquired using a micro endoscope: image courtesy of the Department of Medicine I, Friedrich-Alexander University Erlangen-Nuremberg. It exhibits two properties: 1) The entire image is covered with cells. Therefore, no cell/background separation is required. 2) Cells show a repetitive pattern, and hence, Fourier analysis can be employed for cell detection and/or cell density estimation.

free from illumination artifacts), the change in the detection error was 8% in the worst case.

- Learning is scale- and orientation-invariant.
- Adaptation of some typical computer vision features such as intensity stencils, variance maps, and ray features, from being pixel-based features to adopting a keypoint-based extraction scheme. The advantages of this adaptation are: 1) sparsity, 2) scale- and orientation-invariance, 3) the features can be extracted in a more meaningful way since their related parameters can be tailored according to a relevant scale and orientation.
- The results of the two classification steps used in point-based cell detection (cell/background and cell/cell described in the previous section) were aggregated in a hierarchical clustering framework yielding higher detection accuracy. Moreover, a novel linkage method was suggested which incorporates application-specific information from SIFT keypoints into the linkage method. This linkage was proved to be combinatorial[7] and monotonic. Therefore, it can be computed efficiently and it is also guaranteed to produce clustering trees without reversals. These concepts will be clarified in Chapter 5.

- Local phase and physical phase information were employed for improving supervised cell detection:
  - Improving pixelwise cell/background classification rate using the so-called low-pass monogenic signal framework [Mual 14c].
  - Utilizing the low-pass monogenic signal and the transport of intensity equation in order to achieve better joint learning of suspended and adherent cells [Mual 14a]. Joint learning refers to training a system to detect cells in images which contain both suspended and adherent cells with an accuracy which is comparable to the separate learning case. The latter, i. e. separate learning, refers to the situation when training and testing are done either only on suspended cells or only on adherent cells.

- An unsupervised cell detection algorithm [Mual 14b] which provides an alternative to the supervised approaches in cases where reliability of the detection can be compromised for having a labeling-free system. The main contributions in this part are:
  - A novel self-labeling algorithm for generating ground truth from an input image (test image). This automatically-generated ground truth is used then to train a classifier to separate cells from each other.
  - Employing SIFT for unsupervised structure-of-interest measurements such as mean structure size and dominant curvature direction. One advantage of this point is that the approach parameters can be set safely independent of cell type or image resolution.

[7] The term "combinatorial" is to be understood here in the context of hierarchical clustering rather than combinatorial optimization. A clear definition of this concept is given in Section 3.3.1.

– Good detection accuracy with very short runtime on images of the two most-widely used microscope modalities for unstained imaging: phase contrast microscopy and bright field microscopy.

1.4 Structure of this Work

In Chapter 2, the physical principles of different microscope types are explained. This includes the fundamentals of bright field, phase contrast, and fluorescence microscopy. In addition, quantitative phase microscopy based on the TIE is clarified and major differences to phase contrast microscopy are highlighted. Chapter 3 introduces background knowledge about SIFT keypoints, random forests, and hierarchical clustering. Understanding these concepts is essential for appreciating the contributions of this thesis in the chapters which follow. Chapter 4 describes the image materials used for evaluating the algorithms proposed in this work. In Chapter 5, our supervised cell detection approach is explained in detail. This includes heavy experimental evaluations for testing scale-, orientation-, and illumination-invariance. In Chapter 6, we clarify the relation between physical phase and local phase. We also show how phase information can be used for: 1) improving cell/background classification, 2) improving the joint learning of adherent and suspended cells. Our unsupervised cell detection approach is discussed in Chapter 7. The thesis is concluded with an outlook in Chapter 8 and a summary in Chapter 9.

Chapter 2

Light Microscopy

We perceive the physical world around us using our eyes, but only down to a certain limit. Objects with a diameter smaller than 75 μm cannot be recognized by the naked eye [Murp 02], and due to this reason, they remained undiscovered for the most of human history. Entities which belong to this category include cells (diameter of 10 μm), bacteria (1 μm), viruses (100 nm), molecules (2 nm), and atoms (0.3 nm)[1]. In fact, the importance of these micro/nano entities in almost every aspect of our life cannot be sufficiently appreciated. Microscopes are the tools which enable us to extend our vision to the micro-world and, despite the prefix micro- in the name, to the nano-world, too. This chapter takes the reader through the basic principles of the most widely-used light microscopy techniques, their advantages, and their inherent limitations. Further microscope types such as scanning tunneling microscopes or atomic force microscopes are beyond the focus of this text.

2.1 Image Formation with a Thin Lens

Contents of this section belong to common physical knowledge which can be checked in classical books such as [Pedr 06, Feyn 63]. Consider an object with height h standing at a distance d in front of a converging lens with a focal length $f < d$. Naturally, the lens creates an image of this object. The question then arises as how we can determine the height of the image h' and its distance d' to the lens. From a *geometrical optics* perspective, the image formation process can be described using three simple rules (cf. Figure 2.1):

1. An incident light ray which passes through the optical center O does not suffer any refraction.

2. An incident light ray parallel to the optical axis is refracted passing through the image focal point F'.

3. An incident light ray which passes through the object focal point F is refracted parallel to the optical axis.

[1]The diameter measurements given here are for a blood cell, a typical bacterium, an influenza virus, a DNA molecule, and a uranium atom.

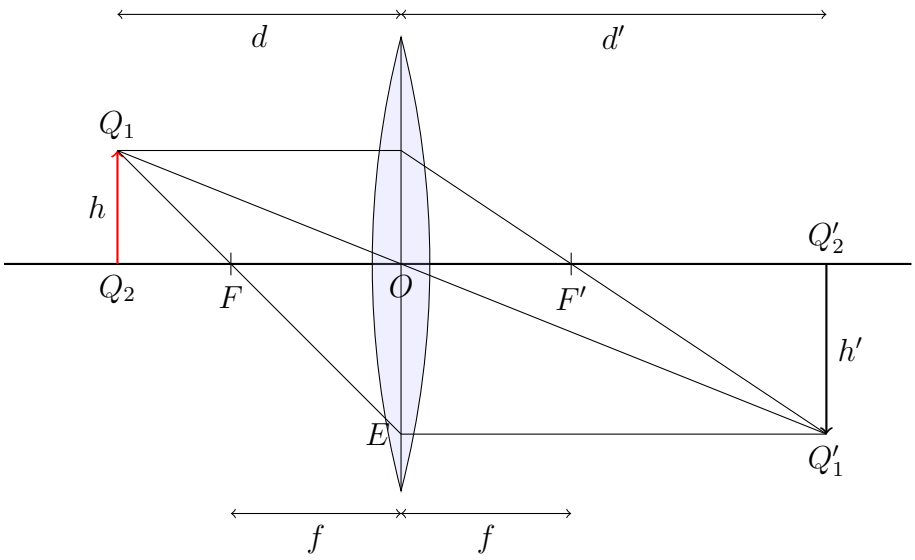


Figure 2.1: Image formation in a converging lens for an object whose distance to the lens is larger than the focal length.

As shown in Figure 2.1, the three rays intersect at a point positioned at distance d' from the lens. Obviously, two rays are sufficient to geometrically construct this intersection point. The image acquired at d' is defined as an *in-focus* image. On the other hand, an image acquired at a longer or a shorter distance than d', is called *defocused* image. In this context, an image of a point source (such as Q_1 in Figure 2.1) is infinitely small at focus (abstracted as a point Q'_1 in Figure 2.1), but it is larger than a point for defocused images.

Figure 2.2 shows the result of applying the rules of image formation, i. e. the three rules mentioned above, on the case when the object is within the focal length ($d < f$). As can be seen in the figure, the rays do not converge. However, the ray extensions intersect at a point Q'_1, called *virtual* image, from which the rays *appear* to diverge. In contrast, the images formed when $d > f$ are called *real* as they are real convergence points of light rays. Virtual images formed by a converging lens are *upright* while the real images are *upside-down*. Another important difference is that virtual images cannot be projected on a screen, a camera film, or any other surface. Nevertheless, they can be perceived by the human eye because the eye behaves as a converging lens which recollects the diverged light rays on the retina.

Figure 2.3 shows the result of applying the rules of image formation in a diverging lens when $d < f$. It should be noted, however, that: contrary to the case of converging lenses, when applying these rules on diverging lenses, the image focus F' is at the side of incident light rays and the object focus F is at the other side of the lens. Similar to the case described in Figure 2.2, the image is upright and virtual. However, in contrast to Figure 2.2, it is demagnified. We obtain this result with a diverging lens when $d > f$ as well.

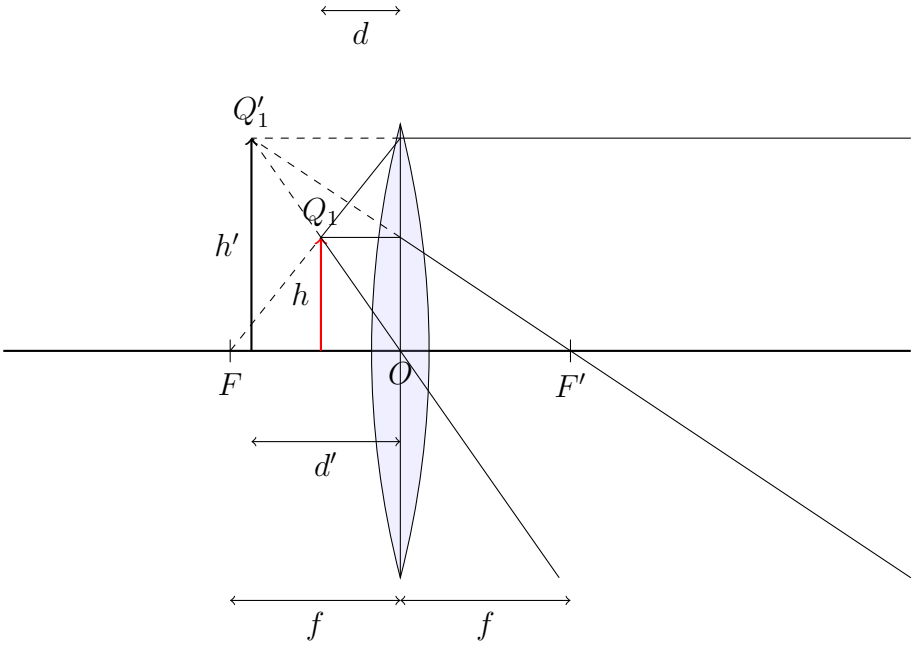


Figure 2.2: Image formation in a converging lens for an object whose distance to the lens is smaller than the focal length.

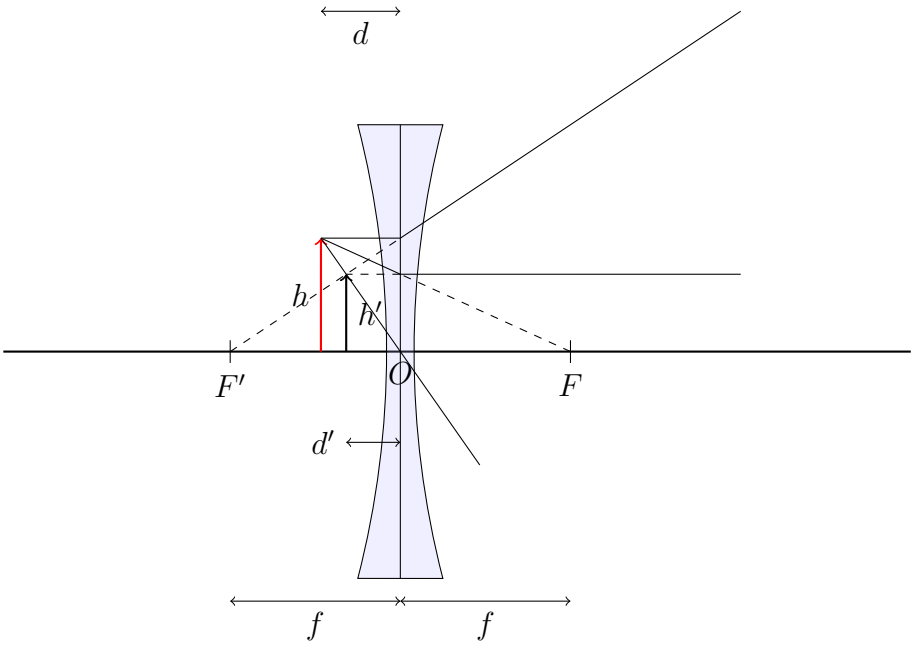


Figure 2.3: Image formation in a diverging lens

Algebraic Formulation

So far we could *geometrically* construct the image of an object in a diverging or a converging lens. At this point, we may ask whether there are closed-form equations which relate the object height h to the image height h', or the object-lens distance d to the image-lens distance d'.

Let us consider a converging lens with $d > f$ (cf. Figure 2.1). From the similar triangles Q_1OQ_2 and $Q'_1OQ'_2$, one can directly write:

$$\frac{h'}{h} = \frac{d'}{d} \tag{2.1}$$

The same applies for triangles Q_1FQ_2 and FOE:

$$\frac{h'}{h} = \frac{f}{d-f} \tag{2.2}$$

Combining Eq. 2.1 and Eq. 2.2 yields:

$$\frac{f}{d-f} = \frac{d'}{d}$$

$$fd = d'd - d'f$$

$$fd + d'f = d'd$$

Dividing by fdd' yields the *thin lens equation*:

$$\frac{1}{d'} + \frac{1}{d} = \frac{1}{f} \tag{2.3}$$

Eq. 2.3 was derived in this text for real images in a converging lens. Nevertheless, it can be also used for virtual images and/or diverging lenses under the following sign conventions: 1) d' is negative when the image is at the object side of the lens (similar to the case in Figure 2.2), otherwise it is positive. 2) f is negative for diverging lenses. Moreover, if we add a third sign convention stating that h' is positive for upright images and negative otherwise, then Eq. 2.1 and Eq. 2.2 can be generalized to the following form:

$$\text{MGN} = \frac{h'}{h} = -\frac{f}{d-f} = -\frac{d'}{d} \tag{2.4}$$

Based on the above-mentioned sign conventions, the magnification MGN is positive for upright images and negative for upside-down images. This generalization, i. e. Eq. 2.3 and Eq. 2.4, can be proved to be correct by applying the three rules of geometric image formation and employing triangle similarity for each specific setup. Moreover, based on Eq. 2.4, the following conclusions can be drawn:

- The image of an object in a converging lens is magnified ($|\text{MGN}| > 1$) when $d < 2f$, has the same size of the object when $d = 2f$, and demagnified ($|\text{MGN}| < 1$) when $d > 2f$.
- The image of an object in a diverging lens ($f < 0$) is demagnified.

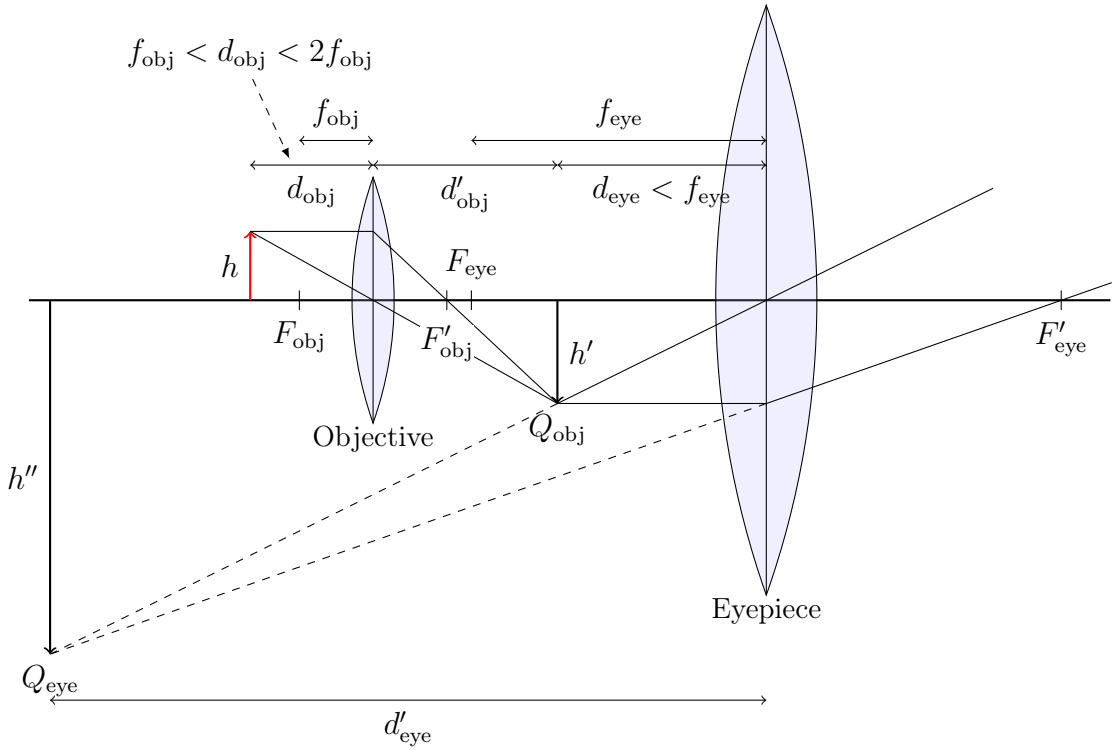


Figure 2.4: Image formation in a compound microscope. Symbols F_{obj}, F'_{obj}, F_{eye}, and F'_{eye} represent the objective object focal point, objective image focal point, eyepiece object focal point, and eyepiece image focal point, respectively. A human observer at the right-hand side of the figure will see the image Q_{eye}.

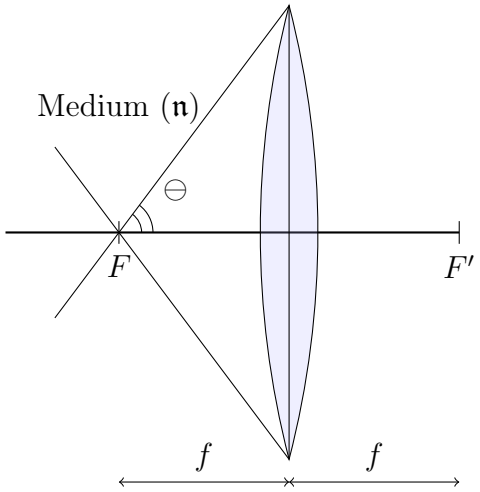


Figure 2.5: The numerical aperture is determined by $\ominus$ the half angle of the maximum light cone and $\mathfrak{n}$ the refractive index of the medium between lens and specimen.

2.2 Compound Microscope

If you look through a magnifying glass at an object located within the focal length of the lens, you see a magnified upright virtual image of the object. Conceptually, this is a simple microscope. The *compound microscope* (cf. Figure 2.4) extends this basic principle by using at least two converging lenses. The lens which is closer to the specimen is called *objective* lens. It creates a real magnified inverted image Q_{obj} of the specimen. This requires that the specimen distance to the objective d_{obj} is in the range $f_{\text{obj}} < d_{\text{obj}} < 2f_{\text{obj}}$, where f_{obj} is the focal length of the objective. The second lens is called *eyepiece* as it is the component through which a user of the microscope observes the sample. The distance of Q_{obj} to the eyepiece d_{eye} is, by construction, less than the focal length of the eyepiece ($d_{\text{eye}} < f_{\text{eye}}$). Consequently, the eyepiece lens creates a magnified virtual image Q_{eye} of Q_{obj}. Since the image of the first lens is an object for the second one, the total magnification is the product of the two lens magnifications [Murp 02].

In modern microscopes, the objective lens is characterized by its magnification and numerical aperture. The magnification was defined above in Eq. 2.4. The numerical aperture NA quantifies the capability of a lens to gather light. It is defined as follows [Wu 08]:

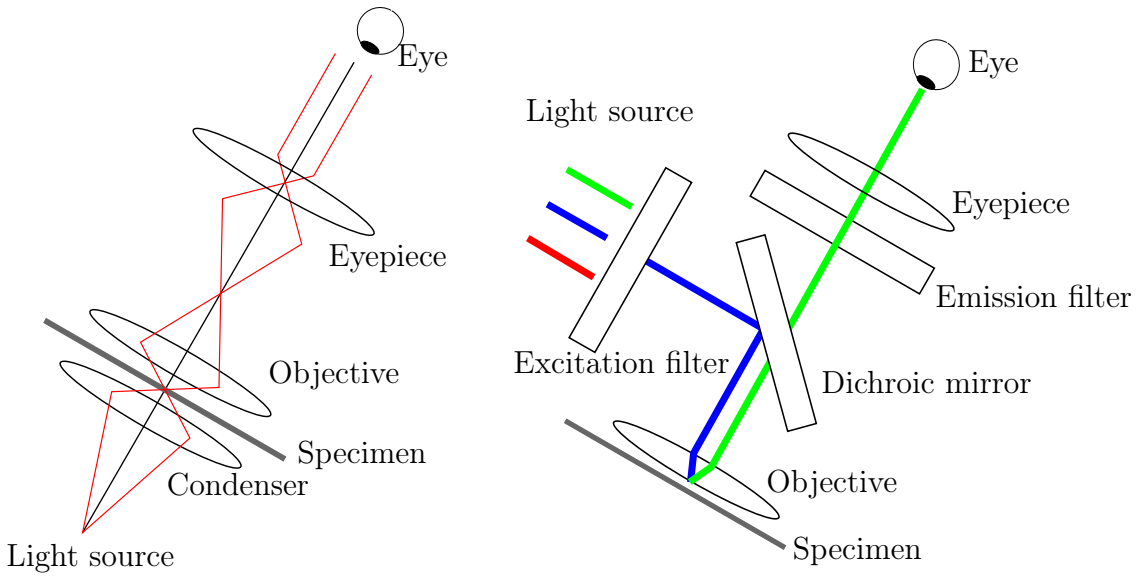
$$\text{NA} = \mathfrak{n} \ \sin \ominus, \tag{2.5}$$

where $\mathfrak{n}$ is the refractive index of the medium between objective lens and specimen ($\mathfrak{n}_{\text{air}} \approx 1$) and $\ominus$ is the half angle of the maximum light cone which the lens can collect (cf. Figure 2.5). Since the image formed by the objective lens is real, it can be captured by a physical detector. For instance, it can be recorded by a CCD chip, and hence, the magnified view can be saved as a digital image which can be further processed by a digital computer.

The principle of compound microscope models the magnification mechanism. Additionally, depending on how the sample is illuminated and which kind of information is carried by light rays, light microscopes can be further classified into subcategories: bright field, fluorescence, phase contrast, quantitative phase, and others. In the following sections, more details will be given about each of the aforementioned microscopic modalities.

2.3 Bright Field Microscopy

Typically, the density and thickness of a specimen are space-variant (change in space). Consequently, specimen points absorb light differently, i. e. the energy of light after passing through the specimen is, likewise, space-variant. Figure 2.6a schematically shows how this fact can be utilized in a microscopic setup. The *condenser* shown in the figure plays the role of concentrating light coming from a light source at the specimen [Albe 05]. The specimen information is encoded in the intensity of light wave which reaches the objective. Background or the part of the scene which does not contain dense objects tends to be bright in the resulting image [Lace 99]. This observer impression gave the technique its name. Bright field setup is the number-one choice whenever minimization of expenditure or implementation difficulties are



(a) Scheme of a bright field microscope (b) Scheme of a fluorescence microscope

Figure 2.6: Basic diagrams of a bright field microscope and a fluorescence microscope. Both were drawn after [Albe 05].

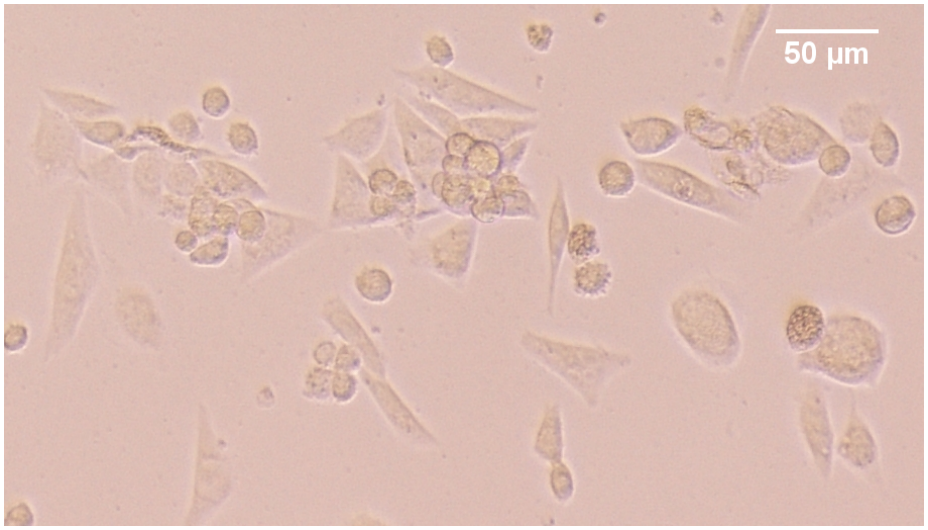
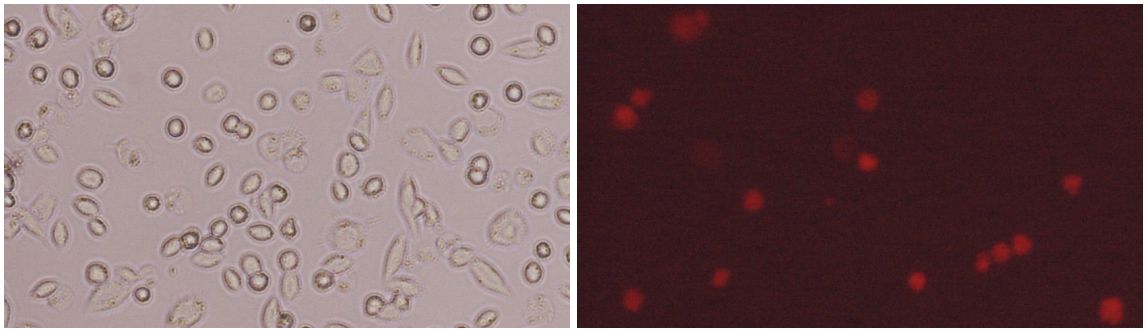


Figure 2.7: A microscopic image of a cell culture: the image was acquired using a Nikon Eclipse TE2000U microscope with a bright field objective of magnification 10× and NA = 0.3.



(a) A bright field image of CHO cells

(b) The same scene at the left-hand side but seen under a fluorescent channel. Red spots indicate dead cells.

Figure 2.8: Illustration of cell viability detection using PI-staining

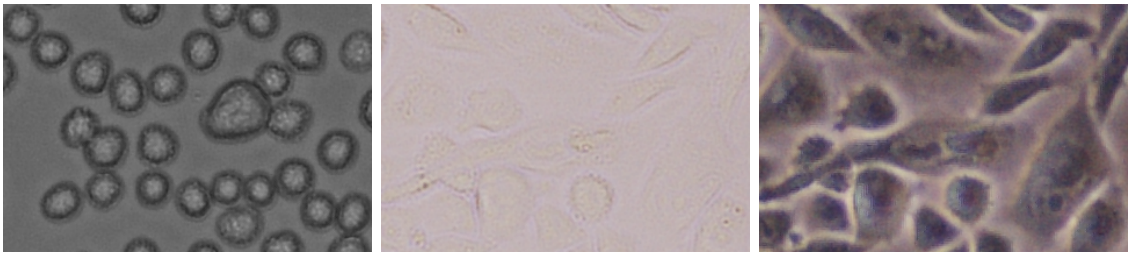
main concerns. Examples of bright field images of cells were shown in Chapter 1. An additional example is shown in Figure 2.7.

2.4 Fluorescence Microscopy

While a bright field microscope utilizes light absorption of a sample, a fluorescence microscope makes use of another natural phenomenon called fluorescence. Some special materials, when illuminated with light having a specific wavelength, emit light with another wavelength. As shown in Figure 2.6b, an excitation filter is required to select a part of the electromagnetic spectrum for exciting the fluorescent materials in specimen. Another filter is then utilized to separate the emitted light from that used in the excitation process.

Fluorescence microscopes deliver images of high contrast when compared to bright field images. In addition, due to the fact that fluorescence can be incited by specific biological or physical processes, scientists were able to find many applications of fluorescence microscopy in materials science and cellular biology. To give just one example, a widely-used technique for cell viability detection (cf. Figure 2.8) is based on imaging of a fluorescent dye called propidium iodide (PI) [Van 13]. Viable cells are usually selectively permeable, i. e. they do not allow molecules to freely cross the cellular membrane. When a cell dies, this exclusion property is lost allowing PI to leak through the cellular membrane toward cell interior. PI binds then to RNA and DNA inside the penetrated cell which drastically enhances the fluorescence [Arnd 89]. Therefore, dead cells can be easily distinguished from the non-stained viable cells.

There are at least two shortcomings of fluorescence imaging: Firstly, staining may cause some undesired effects on the sample under study. For instance, it was shown that the dyes used in cell viability detection affect cell stiffness [Lule 09]. Secondly, what we see under fluorescence microscopy is the activity of fluorescent dyes which, in general, does not reveal structural information. Moreover, these fluorescent dyes do not always cover the entire imaged object [Ali 07]. These two factors lead to incomplete shape information.



(a) A bright field image dominated by amplitude objects: CHO cells in suspension.

(b) A bright field image dominated by phase objects: adherent ultra-thin CHO cells.

(c) A phase contrast image of the same scene shown in 2.9b. In comparison to 2.9b, cells are clearly visible, albeit surrounded by halo artifacts.

Figure 2.9: Examples of amplitude objects and phase objects in biology

2.5 Phase Contrast Microscopy

As mentioned earlier, in bright field microscopy, light absorption is responsible for image formation. Objects which absorb light are called *amplitude objects* since they affect light amplitude. Transparent objects, on the other hand, hardly alter the amplitude of light. They, however, retard light wave introducing a phase shift, and thus, they are given the name *phase objects* [Ager 03]. Typical light detectors such as CCD chips or retina in our eyes can recognize amplitude variations but they are insensitive to phase distortion. In the 1930s, the Dutch physicist Frits Zernike came up with a brilliant trick for converting the invisible phase shift to a visible amplitude change [Zern 55]. His contribution is the basis for a long-established technique in laboratories today known as *phase contrast*. Figure 2.9a shows a bright field image of a sample dominated by amplitude objects. In this particular example, they are cells in suspension. Figure 2.9b also shows a bright field image, but of a sample dominated by phase objects. The sample contains ultra-thin adherent cells. In Figure 2.9c, the same specimen of Figure 2.9b is shown, but under a phase contrast microscope. A considerable improvement in contrast and information content can be clearly seen in the phase contrast image. In the text which follows, in order to grasp a concrete conception of phase and Zernike's trick, we introduce phase in the context of wave equation and thereafter explain the working principle of phase contrast.

2.5.1 Wave Equation

Informally speaking, at a point in space $\mathbf{r} = (x, y, z)^\top$, we can imagine the light activity as a particle dancing in time according to $e^{i\omega t}$, where t is time and ω is the angular frequency which determines light color. In general, this dance is amplitude-scaled and phase-shifted differently at each point in space. Consequently, the wave/particle function $\psi(\mathbf{r}, t)$ can be modeled as follows:

$$\psi(\mathbf{r}, t) = A(\mathbf{r})e^{i(\omega t + \phi(\mathbf{r}))} = A(\mathbf{r})e^{i\phi(\mathbf{r})}e^{i\omega t} = U(\mathbf{r})e^{i\omega t}. \tag{2.6}$$

The term $U(\mathbf{r})$ encodes both amplitude change $A(\mathbf{r})$ and phase shift $\phi(\mathbf{r})$ as a complex number, and thus called *complex amplitude* of the wave. Eq. 2.6 is insufficient to describe a wave unless ψ fulfills the *wave equation* [Good 96]:

$$\frac{\partial^2 \psi}{\partial t^2} = c^2 \nabla^2 \psi, \tag{2.7}$$

where c is the speed of light in the propagation medium, and $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} + \frac{\partial^2}{\partial z^2}$ is the spatial Laplacian. Assuming that ψ can be factorized as $\psi(\mathbf{r}, t) = \psi_{\mathbf{r}}(\mathbf{r})\psi_t(t)$ (which is the case in Eq. 2.6), one can derive the *time-independent wave equation*, also known as *Helmholtz's equation* [Sale 07]. More specifically, using Eq. 2.6 in Eq. 2.7 yields:

$$\begin{aligned} \frac{\partial^2}{\partial t^2}\left(U(\mathbf{r})e^{i\omega t}\right) &= c^2\nabla^2\left(U(\mathbf{r})e^{i\omega t}\right) \\ -\omega^2 U(\mathbf{r})e^{i\omega t} &= c^2 e^{i\omega t}\nabla^2 U(\mathbf{r}) \\ c^2\nabla^2 U(\mathbf{r}) + \omega^2 U(\mathbf{r}) &= 0, \end{aligned}$$

which is typically written in terms of the *wavenumber* k, defined as $k = \frac{\omega}{c}$, in the following form:

$$\nabla^2 U(\mathbf{r}) + k^2 U(\mathbf{r}) = 0. \tag{2.8}$$

An important class of solutions for Helmholtz's equation is given by the following complex amplitude:

$$U_{\mathfrak{l}}(\mathbf{r}) = A_{\mathfrak{l}} e^{i\ \mathbf{k}^\top \cdot \mathbf{r}}. \tag{2.9}$$

In this solution, the amplitude is constant everywhere with a real value $A_{\mathfrak{l}}$ while the phase is linearly dependent on position $\phi_{\mathfrak{l}} = \mathbf{k}^\top \cdot \mathbf{r} = xk_x + yk_y + zk_z$. In order for Eq. 2.9 to satisfy Helmholtz's equation, $\mathbf{k}$ must fulfill $\sqrt{k_x^2 + k_y^2 + k_z^2} = k$ [Sale 07]. This fact can be easily verified by setting $U(\mathbf{r}) = U_{\mathfrak{l}}(\mathbf{r})$ in Eq. 2.8. Moreover, the locus of points in space for which $U_{\mathfrak{l}}(\mathbf{r}) =$ constant, is clearly a plane with a normal vector $\mathbf{k}$. Therefore, waves described by Eq. 2.9 are called *plane waves*.

2.5.2 Phase Contrast Principle

The core concept of Zernike's phase contrast can be appreciated under some simplifications. Consider a plane wave propagating along the z-direction, i. e. $k = k_z$. In addition, consider an image detector positioned at $z = z_d$ perpendicular to the z-axis. In the absence of a specimen, the complex amplitude of incident light at the detector is given by:

$$U_{in} = A_{in} e^{i\ kz_d}, \tag{2.10}$$

where A_{in} is a real constant. U_{in} can be thus seen as a complex constant (see the explanation of Eq. 2.9). The effect of specimen on the incident light can be modeled by a complex multiplication of U_{in} with the specimen transmittance function defined as [Heis 10]:

$$U_f(x, y) = a(x, y)e^{i\phi_{\text{diff}}(x,y)}, \tag{2.11}$$

where $a(x, y)$ and $\phi_{\text{diff}}(x, y)$ are, respectively, the amplitude change and the phase shift introduced by the specimen. Consequently, the presence of a specimen modeled by U_f yields the following complex amplitude at $z = z_d$:

$$U_s(x, y) = U_{in} a(x, y) e^{i\phi_{\text{diff}}(x,y)} = A_{in} a(x, y) e^{i(kz_d + \phi_{\text{diff}}(x,y))}. \tag{2.12}$$

Ideal amplitude objects are characterized by $a \ll 1$ due to the reduction of light amplitude induced by the absorption, whereas $a = 1$ holds for ideal phase objects. The detector at $z = z_d$ is only sensitive to amplitude variations, and hence, from the wave described in Eq. 2.12, the following image will be recorded:

$$I_s(x, y) = |U_s(x, y)|^2 = A_{in}^2 a^2(x, y)\cdot \tag{2.13}$$

This equation clearly reveals that phase objects in the scene ($a = 1$) will be invisible. Moreover, under the assumption of *weak phase objects* ($\phi_{\text{diff}}(x, y) \ll 1$ rad), the complex exponential can be approximated by the first two terms of its Taylor series [Glck 09]. Therefore, Eq. 2.12 yields:

$$U_s^p(x, y) = U_{in}(1 + i\phi_{\text{diff}}(x, y))\cdot \tag{2.14}$$

The constant term 1 can be interpreted as unscattered light, i. e. the part of light wave which is unaffected by the sample. The phase term $\phi_{\text{diff}}(x, y)$, on the other hand, represents the scattered light which carries information about the sample. The image, which the detector records when it receives this wave, is given by:

$$I_s^p(x, y) = |U_s^p(x, y)|^2 = A_{in}^2(1 + \phi_{\text{diff}}^2(x, y)) \approx A_{in}^2, \tag{2.15}$$

which shows again that the phase shift introduced by phase objects cannot be captured by the detector. Let us now consider Eq. 2.14 in the Fourier domain. Applying Fourier transform on both sides of Eq. 2.14 yields:

$$\mathcal{U}_s^p(k_x, k_y) = U_{in}\left((2\pi)^2 \delta(k_x, k_y) + i\Phi_{\text{diff}}(k_x, k_y)\right), \tag{2.16}$$

where $(k_x, k_y)^\top$ is the vector of spatial frequencies[2], and δ is the Dirac delta function. From a mathematical perspective, Zernike's idea can be conceptualized as a shift of the DC component by $\pi/2$ (a multiplication with i) so that the scattered and the unscattered light waves are *in-phase* [Glck 09]:

$$\mathcal{U}_{s,\text{Zernike}}^p(k_x, k_y) = U_{in}\left(i\,(2\pi)^2 \delta(k_x, k_y) + i\Phi_{\text{diff}}(k_x, k_y)\right)\cdot \tag{2.17}$$

At the hardware level, this shift of the DC component is achieved using a flat sheet of glass. By transforming Eq. 2.17 back to the spatial domain, we obtain:

$$U_{s,\text{Zernike}}^p(x, y) = U_{in}\left(i + i\phi_{\text{diff}}(x, y)\right)\cdot \tag{2.18}$$

Consequently, the following image is acquired at the detector:

$$\begin{aligned} I_{s,\text{Zernike}}^p(x, y) = \left|U_{s,\text{Zernike}}^p(x, y)\right|^2 &= A_{in}^2\left(1 + 2\phi_{\text{diff}}(x, y) + \phi_{\text{diff}}^2(x, y)\right) \\ &\approx A_{in}^2\left(1 + 2\phi_{\text{diff}}(x, y)\right)\cdot \end{aligned} \tag{2.19}$$

Comparing Eq. 2.19 with Eq. 2.15 reveals how Zernike's trick converts the phase shift to a visible contrast in the acquired image.

[2]Notation (k_x, k_y) is used as a spatial frequency (Fourier domain) even though it was used earlier in the text to denote a wave vector. This is justified because wave vector has a spatial frequency interpretation [Mir 12]. This fact can be checked by applying Fourier transform on both sides of Eq. 2.8.

2.6 Quantitative Phase Microscopy

In the previous section, phase was employed to obtain more contrast of transparent specimens. At this point, we may ask the following question: what does the numerical value of phase tell us about the physical properties of a specimen? In fact, the phase difference introduced by a phase object (cf. Eq. 2.11) can be given as follows [DiMa 11]:

$$\phi_{\text{diff}}(x,y) = k \int_{z_1(x,y)}^{z_2(x,y)} \Delta\mathfrak{n}\,(x,y,z)\,dz, \tag{2.20}$$

where k is the wavenumber of incident light, $\Delta\mathfrak{n}\,(x,y,z)$ is the difference in refractive index between the object and surrounding medium, z_1 and z_2 are the start and end coordinates of light path through the object. If the object has a homogeneous refractive index, Eq. 2.20 reduces to:

$$\phi_{\text{diff}}^{\text{hom}}(x,y) = k \cdot \Delta\mathfrak{n} \cdot \text{th}\,(x,y)\,, \tag{2.21}$$

where $\text{th}\,(x,y)$ is the object thickness at (x,y). The product of refractive index with the geometric length of light path is usually termed optical path length (OPL). In addition, the difference of two OPL values is called optical path difference (OPD). Therefore, the numerical value of phase is interpreted as OPD between the object and the surrounding medium (the constant k is ignored). Phase contrast (cf. Section 2.5) is convenient for *qualitative* unstained imaging of transparent specimens. However, it is not suitable for obtaining *quantitative* phase values for two reasons: Firstly, phase information is perturbed by artifacts, called *phase halos*, in image regions which surround phase objects (cf. Figure 2.9c) [Curl 04, Heis 10]. Secondly, Eq. 2.19 which links an observed intensity value in a phase contrast image to the corresponding phase value is valid only for very small phase shifts [Glck 09, Heis 10].

Quantitative phase microscopy (QPM) is an umbrella term for a set of techniques by which it is possible to obtain reliable quantitative phase information. In this text, we confine ourselves to discuss one of these methods: the transport of intensity equation TIE. Teague derived the TIE in 1983 [Teag 83] starting from Helmholtz's equation (cf. Eq. 2.8) under the approximation of a slowly varying field along the z-axis:

$$-k\frac{\partial I\,(x,y)}{\partial z} = I\,(x,y) \cdot \nabla_\perp^2 \phi\,(x,y) + \nabla_\perp I\,(x,y) \cdot \nabla_\perp \phi\,(x,y)\,, \tag{2.22}$$

where $I\,(x,y)$ is the at-focus intensity image (related to the complex amplitude in Eq. 2.8 by $I = |U|^2$), and $\nabla_\perp$ is the gradient operator in the lateral directions, i. e. in the xy plane. The symbol ϕ denotes the phase difference (cf. Eq. 2.11 and Eq. 2.20), but ϕ was used instead of ϕ_{diff} as the phase appears only in differential terms in the TIE. In other words, the phase in TIE is defined up to an additive constant which makes no difference between ϕ and ϕ_{diff}. This equation can be further simplified if we assume ideal phase objects, i. e. $I\,(x,y) = \text{constant} = I_0$, to the following form [Ager 03, Mir 12]:

$$-k\frac{\partial I\,(x,y)}{\partial z} = I_0 \nabla_\perp^2 \phi\,(x,y) \cdot \tag{2.23}$$

The axial derivative at the left-hand side of Eq. 2.22 or Eq. 2.23 can be measured:

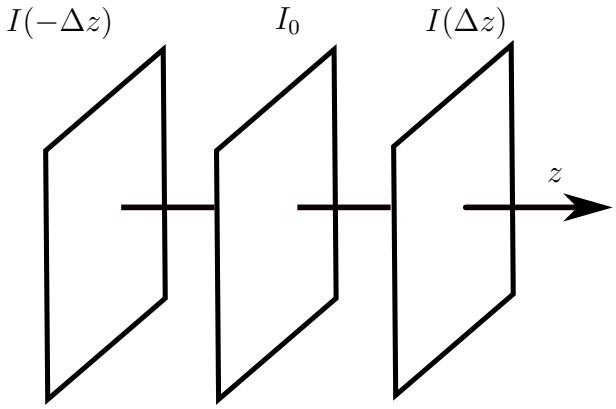
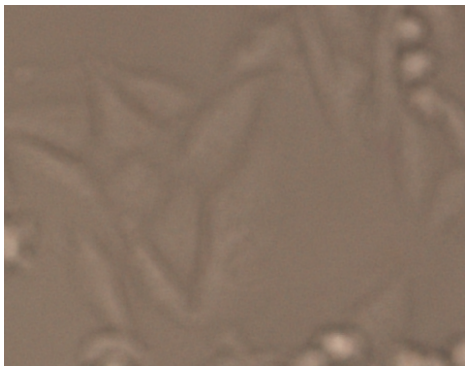
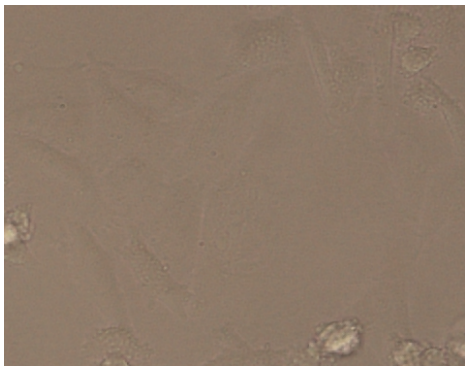


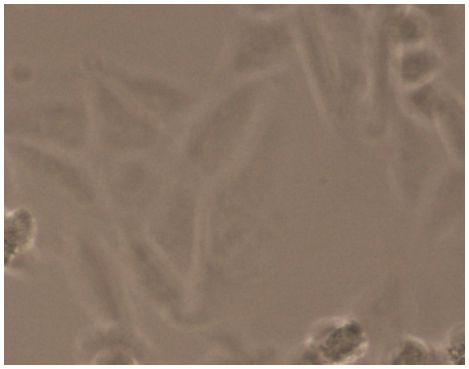
Figure 2.10: The axial derivative of the wave intensity at focus can be measured by subtracting two images at two different focus levels.



(a) A defocused bright field image of the cell culture: $\Delta z = -15$ μm



(b) A bright field image of the cell culture at-focus: $\Delta z = 0$



(c) A defocused bright field image of the cell culture: $\Delta z = +15$ μm

(d) A quantitative phase map obtained by solving the TIE. The bias field was partially corrected using a bias-correction algorithm.

Figure 2.11: Illustration of QPM using the TIE. The figures show a cell culture of adherent ultra-thin L929 cells.

First, acquire a *bright field* image at focus I_0. Defocus the microscope by a distance Δz and acquire another image $I(\Delta z)$ (cf. Figure 2.10). The finite-difference approximation of the derivative is then given by $\frac{I(\Delta z)-I_0}{\Delta z}$. After estimating the axial derivative, the only unknown which is left in the TIE is the phase. Therefore, the TIE can be solved for ϕ yielding a quantitative phase map. Figure 2.11 exemplifies defocused images, an at-focus image, and a TIE solution.

Earlier in this text (cf. Section 2.5.2), it was mentioned that ideal phase objects are invisible in bright field microscopy. As pointed out in Chapter 1 and also demonstrated in Figure 2.11, the aforementioned statement is correct only under the condition that the image is acquired at focus. This phenomenon, i. e. the possibility to visualize phase objects in bright field microscopy, can be interpreted in the light of TIE. The contrast obtained by defocusing is numerically represented by the left-hand side of Eq. 2.23. The right-hand side reveals that this contrast is, in fact, phase information. The employment of defocusing to visualize transparent samples in a bright field setup is sometimes called **defocusing microscopy** [Ager 03].

Due to the quantitative nature of TIE results, it can be utilized to compute specimen physical descriptors which are difficult to obtain using phase contrast. For instance, it can be in principle used for estimating cell thickness and volume in biological cell cultures. In general, TIE seems to be attractive when compared to phase contrast for at least two reasons: 1) It is possible to obtain high-contrast phase images using a bright field microscope which is cheap and easy to implement compared to a phase contrast microscope. 2) TIE yields quantitative rather than qualitative phase information. However, every new technique comes with its own problems, and TIE is by no means an exception to this rule. In fact, estimating the axial derivative is very sensitive to the selection of defocus distance Δz [Paga 04, Wall 10]. In addition, a TIE solution is prone to be perturbed by a low-frequency bias field which needs to be corrected [Ali 10].

2.7 Limitation of Light Microscopy

In Figure 2.1, a *point source* creates a *point image* at focus. This is, however, a result of geometrical optics which does not take the wave nature of light into account. From a wave-optics perspective, light exhibits the properties of waves, and hence, it undergoes diffraction upon encountering a barrier or a slit. In microscopy, this slit is the finite-sized aperture of the objective. Due to the diffraction process, the image of a point source is a pattern known, after Sir George Airy, as *Airy pattern*. As shown in Figure 2.12a, it is composed of a central spot, known as *Airy disk*, surrounded by multiple diffraction rings. The radius of Airy disk, when the image is in its best focus, is [Murp 02]:

$$d_{\text{Airy}} = 0.61\frac{\lambda}{\text{NA}}, \tag{2.24}$$

where $\lambda = \frac{2\pi}{k}$ is the wavelength of incident light, and NA is the numerical aperture (cf. Eq. 2.5). It is noteworthy to mention that d_{Airy} in Eq. 2.24 is given in object-space units. Therefore, in image plane, the radius of Airy disk is $\text{MGN} \cdot d_{\text{Airy}}$, where MGN is the magnification.

The *resolving power* of a microscopic system is defined as the minimum distance between two point sources in the object space for which they are still discernible as two points in the image plane. Intuitively, the two points are distinguishable as long as the sum of the two corresponding Airy patterns contains two distinct peaks. However, the condition under which the two peaks are considered *distinct*, can be defined in several ways. This led to different, but similar, definitions of the resolving power. According to Rayleigh, it is given by the radius of Airy disk $d_{\text{min}} = d_{\text{Airy}}$ (cf. Figure 2.12b). A slightly different definition, known as *Abbe criterion*, is given as $d_{\text{min}} = 0.5\frac{\lambda}{\text{NA}}$. The reader is referred to [Wu 08] for more details about the two aforementioned criteria.

In order to enhance microscopic resolution, one needs to employ light of shorter wavelength and/or an objective of higher numerical aperture. Using shorter wavelengths will be considered in the next section. The numerical aperture, as revealed by Eq. 2.5, is theoretically upper-limited by unity when air ($\mathfrak{n}_{\text{air}} \approx 1$) is the medium between the specimen and the objective. In order to go beyond this limit, microscope manufacturers designed objectives which can function when a medium of higher refractive index such as water ($\mathfrak{n}_{\text{water}} \approx 1.33$) or oil ($\mathfrak{n}_{\text{oil}} \approx 1.51$) is embedded between the specimen and the objective. This led to the development of *water immersion objectives* and *oil immersion objectives*.

If we set wavelength in Eq. 2.24 to the wavelength at the center of visible spectrum $\lambda_{\text{visible}} \approx 550$ nm and numerical aperture to the theoretical upper-bound of oil-immersion numerical apertures $\text{NA}^{\text{best}} = 1.51$, we obtain a Rayleigh resolution of $d_{\text{min}}^{\text{best}} = 222$ nm ≈ 0.2 μm. This value[3] is often cited as the resolution limit of optical microscopy. Two distinct points in object space with distance less than 0.2 μm will be imaged as a sum of two Airy patterns in which only one distinct peak can be recognized. Increasing the magnification will increase the size of this sum of Airy patterns at the image plane, but the enlarged image remains a single-peak pattern. In other words, beyond a certain limit, increasing the magnification does not resolve new details. This phenomenon is known as *empty magnification*.

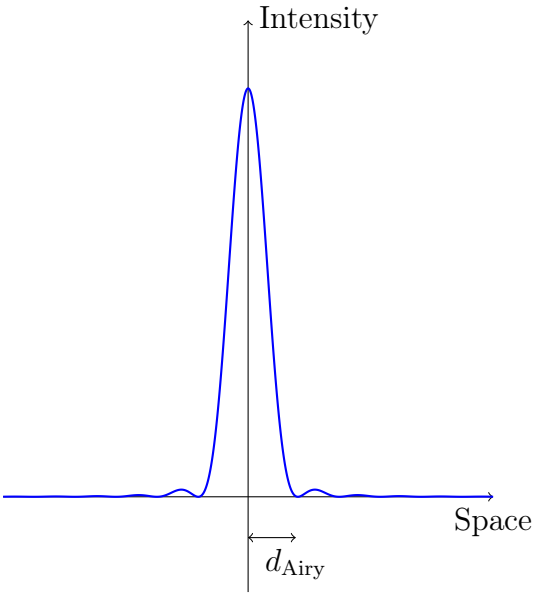
2.8 Beyond Light Microscopy

One obvious way of increasing microscopic resolution is using a wavelength which is shorter than the wavelength of visible light. For instance, it is possible to employ ultraviolet (UV) radiation (wavelength in range 300 - 100 nm), soft X-ray (10 - 1 nm), hard X-ray (below 1 nm)[4], or electron beams (wavelengths below 5 pm are achievable). Each wavelength range allows us to explore a part of the nano-world, but also imposes a new type of challenges for both microscope manufacturers and users.

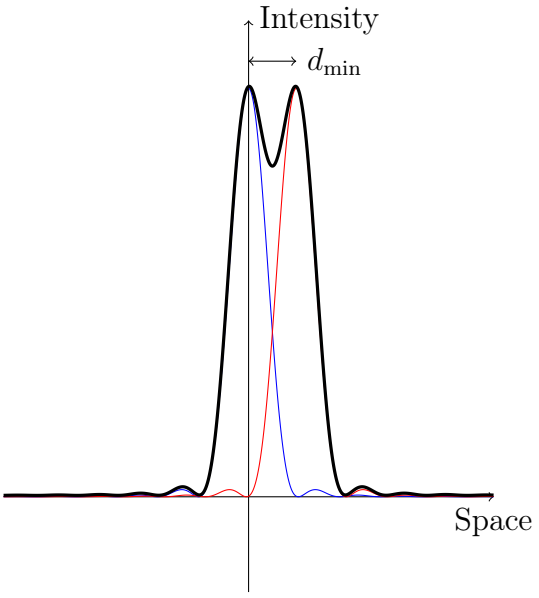
At the UV wavelengths, glass strongly absorbs light radiation, and thus, in **UV microscopy**, the lenses are made of UV-transparent materials such as quartz [Cox 12].

[3] or other close approximations of it depending on the considered upper-limit of numerical aperture and definition of resolving power

[4] X-ray and UV radiation, being a part of the electromagnetic spectrum, belong to *invisible* light. The term light microscopy is, however, restricted to visible light in this text.



(a) Airy pattern composed of Airy disk with radius d_{Airy} surrounded by diffraction rings.



(b) Rayleigh criterion: two features with distance less than $d_{\text{min}} = d_{\text{Airy}}$ will be resolved as a single feature.

Figure 2.12: Diffraction barrier: due to diffraction, the image of a point source is an Airy pattern. The resolving power d_{min} of a microscope is thus limited by the width of this pattern.

Moreover, at the wavelengths of X-ray radiation, the refractive index of solid substances is very close to the refractive index of air. Since the light-focusing performed by a visible-light lens is inherently a refraction process, these lenses cannot be used to focus X-ray beams. In fact, in **X-ray microscopy**, expensive and impractical devices which are based on diffraction instead of refraction are employed to replace the typical optical lenses [Eger 05]. **Electron microscopy (EM)** utilizes electromagnetic lenses and cathode rays in order to achieve a drastic improvement in resolution compared to light microscopy. Unlike ultraviolet and X-ray radiation, cathode rays, being electron beams of measurable mass and negative charge, do not belong to the electromagnetic radiation. Therefore, the photon-wave duality, and hence the conception of wavelength, are not directly applicable. One of the major contributions which led to the development of electron microscopy is the theory of Louis de Broglie who stated in his PhD thesis that the particle-wave duality is also valid for matter. According to de Broglie, the wavelength of an electron of rest mass rm_e and speed c_e is given by [De B 65]:

$$\lambda_e = \frac{\mathfrak{P}}{\mathrm{rm}_e \cdot c_e}, \tag{2.25}$$

where $\mathfrak{P}$ is Planck constant. As an alternative for reflection in optical lenses, in electromagnetic lenses, deflection of electron beams by magnetic fields was exploited to focus the beams. In an electron microscope, similar to a cathode-ray tube, an electron beam is emitted into vacuum by heating the cathode, and then accelerated by applying a voltage between the cathode and the anode. The speed of the electrons,

and hence the wavelength (cf. Eq. 2.25), can be controlled by varying the voltage. The first electron microscopes were very similar from a schematic point of view to bright field microscopes [Eger 05]. The acquired image is based on the specimen absorption of electrons when transmitted into the sample, and hence, they were given the name **transmission electron microscopes (TEM)**. A resolution as high as 0.2 nm [Wils 12] is achieved by the TEM. A major limitation of this scheme, however, is that only very thin samples can be imaged. **Scanning electron microscopy (SEM)** was developed to cope with this difficulty. In SEM, a primary electron beam is focused by an electromagnetic lens on a very small part of the specimen. This primary beam incites the emission of a secondary electron beam. The intensity of this secondary beam is recorded. Afterwards, the primary beam is moved to another part of the specimen, and the same process is applied. This is repeated so that the entire specimen is scanned in a raster pattern and the final image is obtained from the recorded values of the secondary beam intensities [Eger 05, Chan 09]. SEM can be used to image thick samples, even though it captures only the surface details. In addition, the secondary beam is accompanied with X-ray emission characteristic to the material which emitted it [Chan 09]. Therefore, SEM is employed to reveal the chemical composition of specimens. Both SEM and TEM work in a vacuum. Consequently, they can be used only for dead specimens. From this perspective, X-ray and traditional light microscopy are preferred over EM. Although X-ray and electron microscopes provide a considerable improvement of resolution over light microscopes, they are extremely expensive, require large hardware, and mostly involve complicated sample preparation.

2.9 Light Microscopy Beyond the Diffraction Limit

In the past few years, the so-called **superresolution microscopy** [Hell 09] became an active research trend. Today, based on this technology, there are microscopes which achieve a resolving power of about 10 nm [Galb 11]. While this number is inferior to the EM resolution, the breakthrough lies in the fact that this is achieved using visible light. As stated earlier in this text (cf. Section 2.7), the attainable resolution using visible light is limited to 200 nm. May we then conclude that the theory which led to the diffraction limit in light microscopy is flawed? In fact, superresolution microscopy is based on alternatively making fluorescent molecules in a specimen on and off [Marx 13]. Two adjacent fluorescent molecules with a distance less than 200 nm will not be resolved as two points in a superresolution microscope when both of them are turned on simultaneously. However, this will be the case, i. e. they will be resolved as two points, if only one of them is activated at a specific time, and in addition, there is a mechanism to control this activation process. Superresolution microscopy techniques differ in the way in which this on/off switching is implemented. Major technologies in this field today include: stimulated emission depletion (**STED**), reversible saturable optical fluorescence transitions (**RESOLFT**), and stochastic optical reconstruction microscopy (**STORM**).

Chapter 3

SIFT, Random Forests, and Hierarchical Clustering

In this chapter, we introduce background information necessary to understand the rest of this thesis. In Section 3.1, basic principles behind automatic scale selection and the details of SIFT keypoint detection are clarified. Random forests along with concepts such as decision trees and bagging are explained in Section 3.2. Lastly, some aspects of agglomerative hierarchical clustering are briefly discussed in Section 3.3.

3.1 SIFT

3.1.1 Informal Introduction

In the computer vision literature, there are a plenty of approaches which *detect* points *of interest* in an image and then *describe* the local neighborhood of each point, usually using features which capture the visual appearance in the point vicinity. These methods are thus called local feature *detectors* and *descriptors*. The detectors differ in the way in which they define a point of interest while the descriptors differ in the way in which they represent its vicinity. A survey of local feature detectors can be found in [Tuyt 08].

In SIFT, the point of interest is defined as a *blob*. Informally speaking, and considering a one-dimensional image, a blob is an increase of intensity followed by a decrease, or oppositely, a decrease followed by an increase. Mathematically, this behavior is captured by the second derivative which tends to have a high absolute value at the blob's top or bottom [Mual 12]. Whether it is a bottom or a top is, however, determined by the sign of the second derivative. In two-dimensional images, the same principle applies and the second derivative is merely the Laplacian of image intensity.

Since the estimation of derivatives is very sensitive to noise, a low-pass filter is typically applied on an image before computing its derivatives. If we choose a Gaussian kernel for smoothing, the process of computing the m-th derivative of an image is equivalent to convolving this image with the m-th derivative of the Gaussian kernel. For two-dimensional images and a derivative order $m = 2$, the corresponding kernel is commonly called the Laplacian of Gaussian (LoG).

Intuitively, the width of the selected kernel (e. g. the standard deviation of the LoG) has to be *somehow* related to the target structure size. Since the latter, i. e. the size of structures of interest in a given image, is not known a priori, one may employ a *multi-scale* approach: use multiple LoG kernels of different widths. Important questions arise in this context: can we induce the structure size by comparing the filter responses? How to compare the filter responses in a meaningful way? Is it possible to detect the structure orientation?

These questions and others will be addressed in the next sections. At this point, we remind the reader with the concept of Gaussian scale space (GSS). A GSS of an image $I(x, y)$ is defined as:

$$I_{\text{gss}}(x, y, \tau) = I(x, y) * s(x, y, \tau), \tag{3.1}$$

where $s(x, y, \tau)$ is the isotropic Gaussian kernel with variance $\tau = \sigma^2$:

$$s(x, y, \tau) = \frac{1}{2\pi\tau} e^{-\frac{x^2+y^2}{2\tau}}. \tag{3.2}$$

Space in the scale-space representation is obviously the two spatial dimensions x and y while *scale* is the variance τ of the Gaussian kernel [Lind 09].

3.1.2 GSS and Heat Equation

The diffusion of heat in metals is usually modeled by the so-called *heat equation*. This equation states that the temperature change (derivative with respect to time) at each point of the considered metal surface is proportional to the spatial Laplacian of the heat at that point. If one imagines the image as a metal surface, and the intensity at each pixel as a heat value, how would the image look like after the diffusion of intensity for a short time interval ∂t? This analogy is frequently used in the literature of image processing for different purposes such as anisotropic filtering [Weic 98] and scale space representation [Lind 09]. More specifically, the heat equation in this context can be given as [Lind 09]:

$$\frac{\partial L(x, y, t)}{\partial t} = \frac{1}{2}\nabla^2 L(x, y, t), \tag{3.3}$$

where $L(x, y, t)$ is the intensity at time t and position (x, y), and $\nabla^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2}$ is the spatial Laplacian. The constant $\frac{1}{2}$ in Eq. 3.3 is irrelevant for image processing applications, and in principle, can be replaced with any other arbitrary value. However, as it will become clear soon, the value $\frac{1}{2}$ is convenient for further derivations.

Before discussing solutions of Eq. 3.3, we remind the reader with some properties of the Gaussian kernel (cf. Eq. 3.2). Differentiating both sides of Eq. 3.2 with respect to its variance τ yields:

$$\begin{aligned}
\frac{\partial s}{\partial \tau} &= \frac{-1}{2\pi\tau^2} e^{-\frac{x^2+y^2}{2\tau}} + \frac{1}{2\pi\tau}\frac{x^2+y^2}{2\tau^2} e^{-\frac{x^2+y^2}{2\tau}} \\
&= \left(\frac{-1}{\tau} + \frac{x^2+y^2}{2\tau^2}\right) s(x, y, \tau) \\
&= \left(\frac{x^2+y^2-2\tau}{2\tau^2}\right) s(x, y, \tau)\cdot
\end{aligned} \tag{3.4}$$

On the other hand, the first derivative of s with respect to x is given by:

$$\frac{\partial s}{\partial x} = \frac{-x}{\tau} s(x, y, \tau),$$

and the second derivative is:

$$\begin{aligned}\frac{\partial^2 s}{\partial x^2} &= \frac{-1}{\tau} s(x, y, \tau) + \left(\frac{-x}{\tau}\right)\left(\frac{-x}{\tau}\right) s(x, y, \tau) \\ &= \left(\frac{x^2 - \tau}{\tau^2}\right) s(x, y, \tau)\cdot\end{aligned}$$

Consequently, the spatial Laplacian of s, called LoG as mentioned earlier, is given as:

$$\begin{aligned}\nabla^2 s(x, y, \tau) &= \frac{\partial^2 s}{\partial x^2} + \frac{\partial^2 s}{\partial y^2} \\ &= \left(\frac{x^2 + y^2 - 2\tau}{\tau^2}\right) s(x, y, \tau)\cdot\end{aligned} \tag{3.5}$$

One can easily notice from Eq. 3.5 and Eq. 3.4, that $s(x, y, \tau)$ fulfills the heat equation (Eq. 3.3) under the assumption $t = \tau$.

This result can be employed to derive the relation between the GSS and the heat equation. If we differentiate both sides of Eq. 3.1 with respect to τ, we obtain:

$$\begin{aligned}\frac{\partial I_{\text{gss}}(x, y, \tau)}{\partial \tau} &= \frac{\partial}{\partial \tau}\left(I(x, y) * s(x, y, \tau)\right) \\ &= I(x, y) * \frac{\partial s(x, y, \tau)}{\partial \tau} \\ &= I(x, y) * \frac{1}{2}\nabla^2 s(x, y, \tau) \\ &= \frac{1}{2}\nabla^2\left(I(x, y) * s(x, y, \tau)\right) \\ &= \frac{1}{2}\nabla^2 I_{\text{gss}}(x, y, \tau)\cdot\end{aligned} \tag{3.6}$$

In other words, the GSS fulfills the heat equation. This implies two important results:

1. Starting from an image $L(x, y, 0) = I(x, y)$ and applying the diffusion scheme given by Eq. 3.3 until time $t = \tau$ is equivalent to convolving $I(x, y)$ by a Gaussian kernel with variance τ. This justifies the interchangeable use of the two terms time t and scale τ in context of the GSS theory.

2. The LoG can be approximated by subtracting two successive layers in the GSS as this subtraction is the finite difference estimation of $\frac{\partial I_{\text{gss}}(x, y, \tau)}{\partial \tau}$. This approximation of the LoG is called the difference of Gaussians (DoG) and it is used in SIFT for blob detection instead of the LoG.

3.1.3 Automatic Scale Selection

In this section, we address the problem of comparing Gaussian derivatives at multiple scales and employing this comparison for detection of structure size. A general

solution of the heat equation (cf. Eq. 3.3) for one-dimensional images (without loss of generality) can be given by the following equation [Lind 98]:

$$L(x,t) = e^{-\omega_0^2 t/2} \sin(\omega_0 x), \tag{3.7}$$

which is a sinusoid with angular frequency ω_0 and amplitude exponentially decreasing with scale. This result can be approached as a solution of an initial-value problem as follows: consider a single-frequency one-dimensional image $I_{\omega_0} = \sin(\omega_0 x)$. What is the result of convolving it by the Gaussian kernel $s(x,\tau)$? In the Fourier domain, this convolution can be written as multiplication:

$$\begin{aligned}\mathcal{F}(I_{\omega_0}(x))\,\mathcal{F}(s(x,\tau)) &= -i\pi(\delta(\omega-\omega_0) - \delta(\omega+\omega_0))\, e^{-\tau\omega^2/2} \\ &= -i\pi(\delta(\omega-\omega_0) - \delta(\omega+\omega_0))\, e^{-\tau{\omega_0}^2/2},\end{aligned}$$

where ω is the angular frequency of the Fourier transform, δ is the Dirac delta function, and $e^{-\tau\omega^2/2}$ is the Fourier transform of the Gaussian kernel. By transforming back to the spatial domain, we obtain:

$$\sin(\omega_0 x) * s(x,\tau) = e^{-\tau{\omega_0}^2/2} \sin(\omega_0 x)\cdot \tag{3.8}$$

The fact that smoothing a sinusoidal function yields a scaled version of it is expected since sinusoids are eigenfunctions of linear systems. The right-hand side in Eq. 3.8 is the same right-hand side of Eq. 3.7 given $t = \tau$. The message from both equations is that a sinusoid in GSS keeps its frequency fixed, but its amplitude decays exponentially with scale.

Moreover, the m-th order derivative with respect to space is:

$$\frac{\partial^m L(x,t)}{\partial x^m} = \omega_0^m e^{-\omega_0^2 t/2} \sin\left(\omega_0 x + m\frac{\pi}{2}\right)\cdot \tag{3.9}$$

The amplitude of the derivative is clearly always decreasing with scale. In other words, Gaussian derivatives do not achieve optima over scale. This conclusion conforms to the observation that smoothing does not increase derivative amplitude. However, this is not the case with the γ-normalized derivatives introduced by Tony Lindeberg [Lind 98]:

$$D_n(x,t) = t^{m\gamma/2}\frac{\partial^m L(x,t)}{\partial x^m} \quad \text{(by definition)}\cdot \tag{3.10}$$

Combining Eq. 3.10 and Eq. 3.9 yields:

$$D_n(x,t) = t^{m\gamma/2}\omega_0^m e^{-\omega_0^2 t/2} \sin\left(\omega_0 x + m\frac{\pi}{2}\right)\cdot \tag{3.11}$$

Differentiating both sides of Eq. 3.11 with respect to scale yields:

$$\frac{\partial D_n(x,t)}{\partial t} = \left(\frac{m\gamma}{2}t^{m\gamma/2-1} - \frac{\omega_0^2}{2}t^{m\gamma/2}\right) e^{-\omega_0^2 t/2}\omega_0^m \sin\left(\omega_0 x + m\frac{\pi}{2}\right)\cdot \tag{3.12}$$

The root is obtained at:

$$\begin{aligned} \frac{m\gamma}{2}t^{m\gamma/2-1} - \frac{\omega_0^2}{2}t^{m\gamma/2} &= 0 \\ m\gamma t^{-1} - \omega_0^2 &= 0 \\ m\gamma - \omega_0^2 t &= 0 \\ t_{\text{root}} &= \frac{m\gamma}{\omega_0^2}. \end{aligned} \tag{3.13}$$

Eq. 3.13 can be written in terms of wavelength of the sinusoid $\lambda_0 = \frac{2\pi}{\omega_0}$ as:

$$t_{\text{root}} = \sigma_{\text{root}}^2 = \frac{m\gamma}{4\pi^2}\lambda_0^2. \tag{3.14}$$

Therefore, unlike the amplitude of the derivative $\frac{\partial^m L(x,t)}{\partial x^m}$, the amplitude of the γ-normalized derivative attains a stationary point in the scale dimension. Moreover, since wavelength can be understood in terms of structure size, Eq. 3.14 conveys that σ at which the γ-normalized derivative achieves this stationary point is proportional to structure size. Lastly, this stationary point is unique and it maximizes the amplitude of the γ-normalized derivative along the scale dimension [Lind 98].

Let us find the value of the γ-normalized derivative amplitude at the maximizer by setting t in Eq. 3.11 to t_{root} given by Eq. 3.13:

$$\begin{aligned} \max\left(D_n^{\text{amplitude}}\right) &= \left(\frac{m\gamma}{\omega_0^2}\right)^{m\gamma/2} \omega_0^m e^{-m\gamma/2} \\ &= \frac{(m\gamma)^{m\gamma/2}}{e^{m\gamma/2}} \omega_0^{m(1-\gamma)}. \end{aligned} \tag{3.15}$$

This equation conveys that the value of the γ-normalized derivative amplitude at $\gamma = 1$ has a very desirable property: it is independent of the signal frequency, that is to say it is scale-invariant.

3.1.4 SIFT Detector

In order to detect SIFT keypoints in an input image, a GSS (cf. Eq. 3.1) of this image is required. The parameter $\sigma = \sqrt{\tau}$ in Eq. 3.1 is a continuous variable, and hence, its range needs to be discretized. The discretization is performed at logarithmic steps with base 2 where each doubling of σ is termed *octave*. More specifically, the discrete σ values are given as follows:

$$\sigma(j) = \sigma_0 2^{\frac{j}{\mathfrak{S}}}, j = 0 \cdots \mathfrak{S} \cdot \mathfrak{O} - 1, \tag{3.16}$$

where σ_0 is the initial σ, $\mathfrak{S}$ is the number of scales per octave, and $\mathfrak{O}$ is the number of octaves. Setting these parameters appropriately is essential for robust blob detection. Extensive evaluations were performed by Lowe [Lowe 04] in order to achieve this robustness. For instance, he reported that the best keypoint *repeatability* can be achieved with three scales per octave ($\mathfrak{S} = 3$). Repeatability refers to the possibility

of regenerating the same keypoints in a *transformed* image. Several image transformations were considered including rotation, scaling, contrast change, and perturbation with noise.

As mentioned above, the Gaussian standard deviation at the first level of each octave (in Eq. 3.16, $j = l \cdot \mathfrak{S}, l = 1 \cdots \mathfrak{O} - 1$) is doubled compared to its preceding octave (at $l - 1$). Consequently, the maximum image frequency is halved at the beginning of each octave, and hence, we can subsample by a factor of 2 (sample each second pixel) without violating the sampling theorem. Based on this justification, and in order to improve performance, the SIFT algorithm applies subsampling at the beginning of each octave.

It was mentioned in Section 3.1.1 that blobs tend to have high absolute values of the LoG. Additionally, it was shown in Section 3.1.3 that the γ-normalized Gaussian derivatives achieve maximal amplitudes at scales corresponding to structure size. Moreover, for $\gamma = 1$, the derivative value at the optimum in scale is scale-invariant. Based on these facts, SIFT employs the γ-normalized Laplacian with $\gamma = 1$. This operator is given as $\sigma^2\nabla^2$ by setting $m = 2$ and $\gamma = 1$ in Eq. 3.10 (considering two spatial dimensions x and y). We pointed out in Section 3.1.2 that the LoG can be approximated by the DoG. This approximation is used in SIFT so that a γ-normalized DoG is computed. We will refer to it through this thesis as $\mathrm{DoG}_{\gamma 1}$.

SIFT assumes a *keypoint* at (x, y, σ) if the $\mathrm{DoG}_{\gamma 1}$ achieves a local optimum in scale and space. In other words, $\mathrm{DoG}_{\gamma 1}(x, y, \sigma)$ has to be either larger or smaller than all its 26 neighbors in the scale-space. The value of σ at which $\mathrm{DoG}_{\gamma 1}$ attains the aforementioned optimum is, by definition, the *keypoint scale*. Unfortunately, there is no consensus about using the term *scale* in the literature. It may denote either the variance of the Gaussian kernel used for smoothing [Lind 09, Lind 98] or the standard deviation of this kernel [Lowe 04, Bay 08, Lind 98]. In this work, and in order to avoid any possible ambiguity, we put the following conventions: 1) When it makes a difference, we use symbols σ for the standard deviation and τ for the variance. 2) If no symbols are used, *scale* refers to the variance as defined in Eq. 3.2 while *keypoint scale* refers to the standard deviation. Please note that, in many cases, both concepts are applicable. One last note about terminology: in order to simplify notation, and unless otherwise mentioned, we use (x, y, σ) both for an arbitrary coordinate in scale-space and for a detected keypoint, i. e. for a point in scale-space at which $\mathrm{DoG}_{\gamma 1}$ has a local optimum.

In order to assign an orientation to a keypoint (x, y, σ), SIFT polls orientations of the spatial gradients of $L(., ., \sigma)$ in the vicinity of (x, y). An isotropic Gaussian window with standard deviation 1.5 σ defines the polling region. A histogram of gradient orientations is then computed. The contribution of each gradient vector to the histogram is weighted by the Gaussian window and the gradient magnitude. The orientation ϑ at which the histogram achieves its maximum is considered the keypoint orientation. Basically, it can be interpreted as the dominant gradient orientation in the keypoint neighborhood. In some symmetric structures, this dominant orientation is poorly defined. SIFT addresses this problem by detecting histogram maxima which are not smaller than 80% of the highest histogram peak. For each of them, a new keypoint is created with the same (x, y, σ), but having a different orientation. Lowe reported an improvement in matching stability by considering these

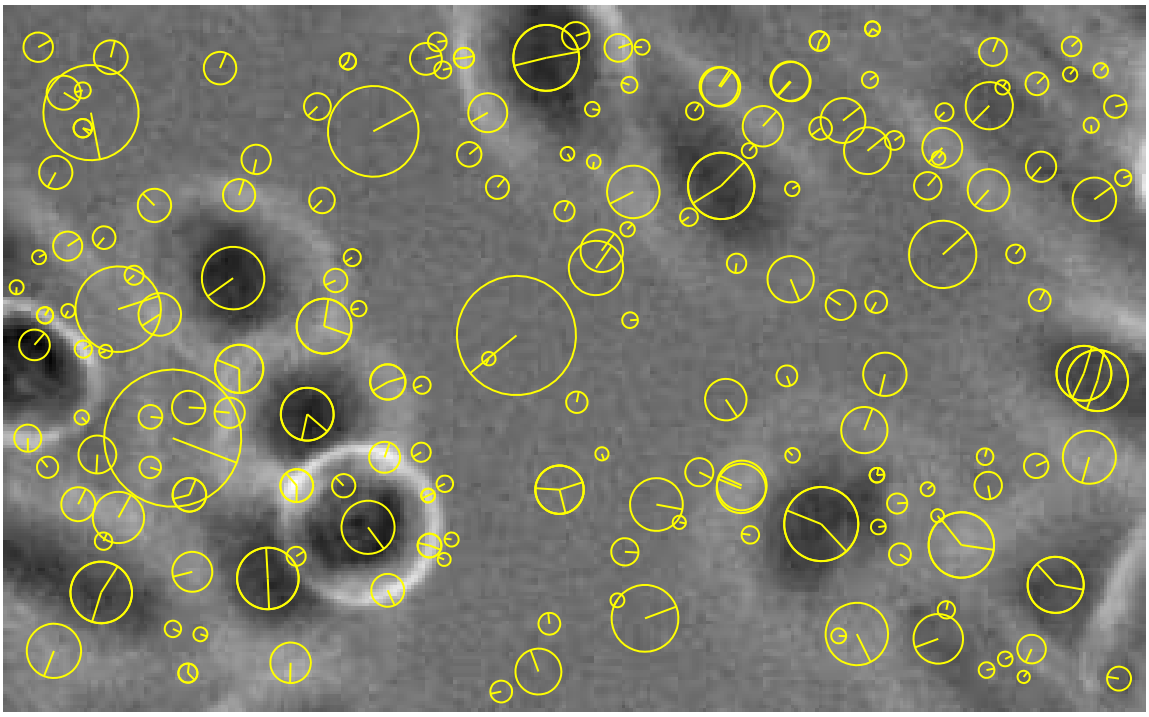


Figure 3.1: Demonstration of SIFT keypoints: typically, each keypoint $(x, y, \sigma, \vartheta)$ is represented by a circle centered at (x, y) with radius equal to σ. The direction of line segment which represents this radius is given by ϑ. Note that some keypoints have multiple orientations.

multiple orientations. SIFT matching refers to finding similarities between images by comparing their SIFT descriptors. Figure 3.1 demonstrates SIFT keypoints on a cell-image example.

The obtained spatial coordinates of SIFT keypoints are real values. In addition, the σ values for these keypoints are not necessarily the discretized σ values given by Eq. 3.16. This is due to the subpixel/subscale interpolation performed by the SIFT algorithm. In order to improve localization in space and scale, the $\mathrm{DoG}_{\gamma 1}(x, y, \sigma)$ at a keypoint location is interpolated as a quadratic function expressed by the first three terms of the Taylor series of $\mathrm{DoG}_{\gamma 1}(x, y, \sigma)$. The local derivatives of $\mathrm{DoG}_{\gamma 1}(x, y, \sigma)$ are estimated at the keypoint location in order to find Taylor coefficients. Afterwards, the extremum of the quadratic is found by setting its derivative to zero and solving the resulting equation. The scale-space location of this extremum is the refined keypoint which is returned by the algorithm. Moreover, the value of the extremum is the refined $\mathrm{DoG}_{\gamma 1}$ value also returned by the algorithm.

SIFT offers a measure which describes circularity of structure. It is the principal curvatures ratio (PCR) defined as:

$$\mathrm{PCR} = \frac{\mathrm{Tr}^2\left(\mathbf{H}_{\mathrm{DoG}_{\gamma 1}}\right)}{\mathrm{Det}\left(\mathbf{H}_{\mathrm{DoG}_{\gamma 1}}\right)}, \tag{3.17}$$

where $\mathbf{H}_{\mathrm{DoG}_{\gamma 1}}$ is the Hessian of $\mathrm{DoG}_{\gamma 1}$ in the xy plane, Tr is the trace, and Det is the determinant. Assuming that the eigenvalues of $\mathbf{H}_{\mathrm{DoG}_{\gamma 1}}$ are $\mathfrak{e}_1$ and $\mathfrak{e}_2$, we can define the following eigen ratio:

$$\mathfrak{e}_r = \frac{\mathfrak{e}_2}{\mathfrak{e}_1},$$

where $|\mathfrak{e}_2| \geq |\mathfrak{e}_1|$. Since there is a maximum or a minimum of $\mathrm{DoG}_{\gamma 1}$ at the keypoint, the two eigenvalues have the same sign, and hence $\mathfrak{e}_r \geq 1$. Accordingly, the PCR can be written as:

$$\begin{aligned}
\frac{\mathrm{Tr}^2\left(\mathbf{H}_{\mathrm{DoG}_{\gamma 1}}\right)}{\mathrm{Det}\left(\mathbf{H}_{\mathrm{DoG}_{\gamma 1}}\right)} &= \frac{\left(\mathfrak{e}_1 + \mathfrak{e}_2\right)^2}{\mathfrak{e}_1 \mathfrak{e}_2} \\
&= \frac{\left(\mathfrak{e}_1 + \mathfrak{e}_r \mathfrak{e}_1\right)^2}{\mathfrak{e}_1 \mathfrak{e}_r \mathfrak{e}_1} \\
&= \frac{{\mathfrak{e}_1}^2 + {\mathfrak{e}_r}^2 {\mathfrak{e}_1}^2 + 2{\mathfrak{e}_1}^2 \mathfrak{e}_r}{{\mathfrak{e}_1}^2 \mathfrak{e}_r} \\
&= \frac{1 + {\mathfrak{e}_r}^2 + 2\mathfrak{e}_r}{\mathfrak{e}_r} \\
&= \frac{\left(1 + \mathfrak{e}_r\right)^2}{\mathfrak{e}_r}.
\end{aligned} \tag{3.18}$$

This function, and hence PCR, has a minimum of 4 when $\mathfrak{e}_r = 1$ which conforms to isotropic blobs. On the other hand, its value increases theoretically until $+\infty$ by increased $\mathfrak{e}_r$ values which conform to increased blob anisotropy.

3.1.5 SIFT Descriptor

After detecting keypoints, the SIFT algorithm creates a *descriptor* for each keypoint $(x, y, \sigma, \vartheta)$ in order to characterize the keypoint vicinity. For feature extraction, the algorithm considers a square neighborhood centered at the keypoint location (x, y) with size proportional to the keypoint scale σ in the pyramid level which corresponds to this scale, i. e. $L(., ., \sigma)$. In the SIFT implementation used in this thesis [Veda 08], the side length of the aforementioned square neighborhood is $4M\sigma$ pixels, where M is a magnification factor which can be determined by user. The neighborhood is divided into 16 subregions with subregion's side length equal to $M\sigma$. For each of them, a histogram of gradient orientations with 8 bins is computed. These orientation histograms form the SIFT descriptor with length $16 \times 8 = 128$ features. Similar to the dominant orientation computation in Section 3.1.4, the contribution of a gradient vector is weighted by the gradient's magnitude and also according to a Gaussian window. The gradient orientations are computed with respect to ϑ and the descriptor coordinates are also rotated relative to ϑ in order to make the descriptor rotation-invariant. Scale-invariance of the descriptor, on the other hand, is a result of choosing a neighborhood's size proportional to the keypoint scale. In Chapter 5, we employ and extend these principles in order to make other feature sets scale- and orientation-invariant.

3.2 Random Forests

The random forest [Brei 01] is a tree-based classifier introduced by Breiman in 2001. It combines three techniques: classification and regression trees (CART), bootstrap aggregation, and random selection of features at each tree's node. These concepts are clarified in the next sections.

3.2.1 CART Trees

A tree classifier or regressor is a hierarchical structure of decision-making nodes which lead to final decisions (class labels or regression values) at the tree leaves. There are many algorithms in the machine learning literature which address the problem of automatically building a decision tree based on training data. Just to name a few: CART [Brei 84], ID3 [Quin 86], C4.5 [Quin 93], and VFDT [Domi 00]. In this section, we consider the CART algorithm, as it is the building-block of random forests.

In order to grow a CART tree for classification on a training dataset TD, the following recursive procedure is followed:

1. Start with the entire training dataset $\mathrm{TD}' = \mathrm{TD}$.
2. If TD' is *pure* enough (see below for definition of purity), stop here (see below for other stopping criteria), mark the current node as *leaf*, and assign the majority-vote class label (most frequent label) in TD' to this leaf. Otherwise, i. e. if TD' is not pure enough, pick the feature $feat$ for which it is possible to find a threshold $thresh$ that can split TD' into two *complementary* datasets TD'_1 and TD'_2 so that the sum of their impurity is minimized. TD'_1 may thus contain feature vectors for which $feat >= thresh$ while TD'_2 contains feature vectors for which $feat < thresh$.
3. Set $\mathrm{TD}' = \mathrm{TD}'_1$ and go to step 2.
4. Set $\mathrm{TD}' = \mathrm{TD}'_2$ and go to step 2.

An impurity measure should quantify the degree of inhomogeneity. For instance, considering a binary classification problem, a dataset with all feature vectors having the label of class 1 is more pure than a dataset with 20% of feature vectors belonging to class 1 and 80% belonging to class 2. The maximum impurity is attained when the two classes have the same relative frequency in the considered dataset, i. e. 50% of the feature vectors belonging to each class. In CART, the two following impurity measures are frequently employed for classification problems [Hast 09]:

$$\text{Entropy}\,(\mathrm{TD}') = -\sum_{j=1}^{N_{\text{Classes}}} \widehat{P}_j \log \widehat{P}_j, \tag{3.19}$$

where N_{Classes} is the number of classes and $\widehat{P}_j$ is the empirical probability (relative frequency) of class j in TD'.

$$\text{Gini index}\,(\mathrm{TD}') = \sum_{j=1}^{N_{\text{Classes}}} \widehat{P}_j(1 - \widehat{P}_j)\cdot \tag{3.20}$$

Decision tree learning algorithms tend in general to build large tress which overfit training data. Moreover, due to its hierarchical structure, the consequences of an over-fitted decision at higher levels propagate to the lower-level nodes [Hast 09]. A small change in training data may thus lead to a substantial difference in the resulting tree structure. This variance reduces the stability and interpretability of decision trees. Some typical procedures are followed in order to lessen the over-fitting effects. For instance, the tree-growing algorithm can be stopped when the cardinality of TD' is below a certain threshold. Moreover, *pruning* is commonly employed in order to restructure a tree and discard its unnecessary splits. Basically, pruning algorithms remove subtrees from the original tree in a way which minimizes some criterion, e. g. a combination of test error and tree size. In the following sections, more effective anti-overfitting techniques such as bootstrap aggregation and random forests will be discussed.

3.2.2 Random Forest Approach

Instead of growing a single decision tree, N_{Tr} trees are grown in the random forest approach [Brei 01]. Each tree is trained on a bootstrap sample of the training data. A bootstrap sample of a dataset TD of size $|\mathrm{TD}| = N$ is obtained by randomly sampling N feature vectors from TD *with replacement*. The probability of a specific feature vector to be outside the resulting sample is $\left(\frac{N-1}{N}\right)^N$. Consequently, the probability that this feature vector belongs to the bootstrap sample is:

$$\widehat{P}_{\mathrm{bootstrap}} = 1 - \left(\frac{N-1}{N}\right)^N.$$

For large datasets, one can write:

$$\begin{aligned}\widehat{P}^{\infty}_{\mathrm{bootstrap}} &= \lim_{N\to+\infty} 1 - \left(\frac{N-1}{N}\right)^N \\ &= \lim_{N\to+\infty} 1 - \left(1 - \frac{1}{N}\right)^N \\ &= 1 - \frac{1}{e} \approx 0.632\cdot\end{aligned}$$

Therefore, in a statistical sense, each bootstrap replicate will contain approximately 63% of the data. That is to say, each tree is trained on a random subset of the feature vectors containing 63% of the entire training set. For a test feature vector, the final decision made by the forest is the majority-vote of all trees in classification and the arithmetic average in regression. In [Brei 96], it was shown that aggregating classifier or regressor outputs trained on bootstrap replicates substaintially improves the accuracy when the individual classifiers or regressors are instable. As mentioned in the previous section, this instability is an inherent property of decision trees. Breiman coined the term **b**ootstrap **agg**regat**ing** or baaging for this procedure.

The random forest employs the bagging concept for building a forest of CART decision trees. However, it goes one step further by injecting a second level of randomness at each tree's node. It modifies the tree's growing algorithm explained in

Section 3.2.1 as follows: at each node, a random subset of N_{Rand} features is selected, and the training algorithm chooses then the best feature from this subset instead of the full feature set. The number of randomly selected features at each tree's node N_{Rand} is a parameter of the training algorithm. Due to the two aforementioned levels of randomness, random forests do not overfit even though the trees are grown with full length without pruning.

The accuracy of random forest is dependent on the strength of each tree, but also on the diversity of search spaces explored by different trees. Large N_{Rand} values push the training algorithm to generate trees which are, more or less, correlated. On the other hand, very small values of N_{Rand} reduce the strength of individual trees. In both cases, the prediction accuracy of random forest degrades. A proper value of N_{Rand} is thus essential to control the trade-off between individual tree strength and inter-tree correlation.

In general, a training procedure of a machine learning model yields a trained model instance and also its mis-prediction error on training data. Since generalization error on unseen data is what matters most, cross-validation or other data sampling techniques are commonly employed to estimate the test error of the model. With random forests, this is not necessary as the test error can be obtained directly from the training algorithm. This internal test error is computed as follows: each feature vector is classified using a sub-forest containing only the trees which did not use this feature vector for training. Statistically, the trees involved in this estimation are about $100\% - 63\% = 37\%$ of the random forest trees. The majority-vote of this sub-forest is considerd to be the prediction result. After applying this process on the entire dataset, the so-called *out-of-bag* (OOB) test error is obtained by calculating the ratio of feature vectors which were misclassified by their corresponding sub-forests.

Another practicality of random forest is that it provides feature ranking in a straightforward manner: in order to rank a specific feature, permute its values randomly in the OOB data of each tree and compute the OOB error of the entire forest after this permutation. The increase in OOB error is a measure of importance of the considered feature. Interested readers are referred to [Brei 02] for more measures of feature importance using random forests.

The imbalance problem is tackled differently in different classifier models. One approach to deal with imbalanced classes in random forests is the *balanced random forest* [Chen 04]: from each class, consider a bootstrap sample of size N_{Minority} which is the number of feature vectors in the minority class. In this way, class labels are uniformly distributed in the bootstrap replicate used for training.

In the last decade, random forest gained a considerable interest due to its desired properties. In a nutshell: it can be used for classification, regression, and clustering with efficient training time and without data normalization. Additionally, and as byproducts, it yields OOB test error estimates and feature ranking. Its accuracy is not very sensitive to its parameters. The parameter which has the greatest impact is N_{Rand}. However, in most cases, default values such as $\log_2(N_F) + 1$ (casted to integer) [Brei 01, Khos 07] or $N_F/5$ [Khos 07], where N_F is the number of features, work pretty well.

3.3 Agglomerative Hierarchical Clustering

Consider a dataset for which a distance measure π between feature vectors is assumed. Evaluations of this measure on all feature-vector pairs in the considered dataset can be written as a matrix W where $W(j,l) = \pi_{jl}$. Conceptually, we can consider each feature vector to be a single-element cluster or a *singleton* cluster [Hast 09]. In order to find clusters at higher levels, agglomerative hierarchical clustering (AHC) in its simplest version conducts the following straightforward algorithm:

1. Start with the initial distance matrix between singleton clusters $W' = W$.
2. Find the minimum value in W'.
3. Merge the two corresponding clusters into a new one.
4. Delete or invalidate entries of the two merged clusters in W' and add an entry of the newly-born cluster.
5. If the algorithm ends up with a single cluster (the entire dataset), stop. Otherwise, go to step 2.

In step 4, adding a new entry representing the emerging cluster requires computing distances between this cluster and each other cluster available at the current iteration. Therefore, a mechanism is needed to induce distances between clusters starting from the available singleton-cluster distances given in W. This mechanism is termed *linkage method* in the hierarchical clustering literature. People in the machine learning community have been using several standard linkage methods for decades. The most common ones include *single*, *complete*, *average*, and *centroid*. In single linkage, the distance between two clusters $\mathbb{A}$ and $\mathbb{B}$ is defined as the minimum distance between $e_j \in \mathbb{A}$ and $e_l \in \mathbb{B}$:

$$\Pi_{\text{single}}(\mathbb{A},\mathbb{B}) = \min_{e_j \in \mathbb{A}\ e_l \in \mathbb{B}} \pi\left(e_j, e_l\right). \tag{3.21}$$

Complete linkage is defined similarly, but using maximum instead of minimum:

$$\Pi_{\text{complete}}(\mathbb{A},\mathbb{B}) = \max_{e_j \in \mathbb{A}\ e_l \in \mathbb{B}} \pi\left(e_j, e_l\right). \tag{3.22}$$

On the other hand, average and centroid linkages adopt more moderate strategies compared to single and complete linkage methods. Average linkage is defined as follows:

$$\Pi_{\text{average}}(\mathbb{A},\mathbb{B}) = \frac{1}{|\mathbb{A}|\,|\mathbb{B}|} \sum_{j=1}^{|\mathbb{A}|} \sum_{l=1}^{|\mathbb{B}|} \pi(e_j, e_l), \tag{3.23}$$

where $|\cdot|$ refers to cluster cardinality. In centroid linkage, the distance between two clusters is defined as the squared Euclidean distance between the two cluster centroids. A cluster centroid is the arithmetic average of feature vectors belonging to the cluster.

Obtaining final clusters, i. e. outputs of the clustering procedure, is typically done by imposing a threshold in Π values so that only the clusters which were formed under this level are considered. The value of the threshold is, however, application-dependent. Therefore, it is desirable to use interpretable distance measures π and linkage methods Π in order to select this threshold in a meaningful way.

Linkage	β_1	β_2	β_3	β_4
Single	$1/2$	$1/2$	0	$-1/2$
Complete	$1/2$	$1/2$	0	$1/2$
Average	$\frac{\lvert\mathbb{A}\rvert}{\lvert\mathbb{A}\rvert+\lvert\mathbb{B}\rvert}$	$\frac{\lvert\mathbb{B}\rvert}{\lvert\mathbb{A}\rvert+\lvert\mathbb{B}\rvert}$	0	0
Centroid	$\frac{\lvert\mathbb{A}\rvert}{\lvert\mathbb{A}\rvert+\lvert\mathbb{B}\rvert}$	$\frac{\lvert\mathbb{B}\rvert}{\lvert\mathbb{A}\rvert+\lvert\mathbb{B}\rvert}$	$\frac{-\lvert\mathbb{A}\rvert\cdot\lvert\mathbb{B}\rvert}{(\lvert\mathbb{A}\rvert+\lvert\mathbb{B}\rvert)^2}$	0

Table 3.1: Coefficients of the Lance-Williams model for some standard linkage methods

3.3.1 Lance Williams Update Formula

In order to compute linkage distances efficiently at a specific iteration, one needs to reuse cluster distances computed during preceding iterations rather than recomputing these distances at each iteration from scratch (e. g. using Eq. 3.21, Eq. 3.22, or Eq. 3.23). In this regard, a linkage method is considered *combinatorial* under the following condition: if cluster $\mathbb{AB}$ is resulting from merging clusters $\mathbb{A}$ and $\mathbb{B}$, then the cluster distance between $\mathbb{AB}$ and any other cluster $\mathbb{C}$ can be induced using the Lance Williams recurrence formula. This formula is given as [Lanc 66, Lanc 67]:

$$\Pi(\mathbb{AB},\mathbb{C}) = \beta_1\Pi(\mathbb{A},\mathbb{C}) + \beta_2\Pi(\mathbb{B},\mathbb{C}) + \beta_3\Pi(\mathbb{A},\mathbb{B}) + \beta_4|\Pi(\mathbb{A},\mathbb{C}) - \Pi(\mathbb{B},\mathbb{C})|, \quad (3.24)$$

where β_1, β_2, β_3, and β_4 are the model coefficients which are characteristic for the linkage method. Table 3.1 shows the coefficient values for some standard linkage methods [Murt 12].

3.3.2 Monotonicity

The graphical representation of an AHC tree is called *dendrogram* where a cluster $\mathbb{AB}$ resulting from merging $\mathbb{A}$ and $\mathbb{B}$ is represented as a node with two daughters, one for each cluster. The height of the node $\mathbb{AB}$ is proportional to $\Pi(\mathbb{A},\mathbb{B})$. In the so-called *monotonic* clustering strategies, $\Pi(\mathbb{A},\mathbb{B})$ is guaranteed to be larger than $\Pi(\mathbb{A}_1,\mathbb{A}_2)$ and $\Pi(\mathbb{B}_1,\mathbb{B}_2)$, where $\mathbb{A}_1$ and $\mathbb{A}_2$ are the two clusters which were merged to form $\mathbb{A}$, while $\mathbb{B}_1$ and $\mathbb{B}_2$ are the clusters which were merged to form $\mathbb{B}$. Non-monotonic dendrograms are usually undesired as they pose difficulty in interpretation [Murt 85, Morg 95]. In fact, you cannot draw a dendrogram as a tree if a *reversal*, i. e. a non-monotonic growing of Π values, occurred during agglomeration. When β_4 is zero, the model given in Eq. 3.24 is monotonic under the following condition [Lanc 67]:

$$\beta_1 + \beta_2 + \beta_3 \geq 1\cdot \quad (3.25)$$

Using Inequality 3.25 and Table 3.1, one can directly conclude that average linkage is monotonic while the centroid linkage is not. Inequality 3.25 cannot be applied for single and complete linkage methods because $\beta_4 \neq 0$. There are, however, more general statements which can be used in this regard. According to [Bata 81], a linkage method is monotonic *iff* all the following conditions are met:

1. $\beta_1 + \beta_2 \geq 0$.

2. $\beta_1 + \beta_2 + \beta_3 \geq 1$.

3. $\beta_4 \geq -\min(\beta_1, \beta_2)$.

Obviously, these constraints are fulfilled with the single and complete linkage methods, and they are thus monotonic.

Chapter 4

Cell Image Materials

In this chapter, we describe image materials which are used for experimental evaluations in the next chapters. For the assessment of our algorithms, cell cultures of different cell lines were cultivated by our project partners at the Institute of Bioprocess Engineering. Since bright field microscopy is the main focus of this thesis, standard bright field images of the cultivated cell lines were acquired. Moreover, a software package for cell image simulation was employed in order to obtain additional images equipped with ground-truth masks. The entire database of both simulated and real standard bright field images is available online[1] so that other researchers can also benefit from it.

Moreover, COSIR images were obtained during the development of COSIR hardware and used for qualitative evaluations. In addition to the bright field images acquired by us, phase contrast images from other research groups were also utilized to evaluate both the supervised and unsupervised cell detection approaches.
We point out that considerable parts of this chapter were already published in:
F. Mualla, S. Schöll, B. Sommerfeldt, A. Maier, and J. Hornegger. "Automatic cell detection in bright-field microscope images using SIFT, random forests, and hierarchical clustering". *Medical Imaging, IEEE Transactions on*, Vol. 32, No. 12, pp. 2274–2286, December 2013.

4.1 Standard Bright Field

Table 4.1 shows a summary of the image sets while Figure 4.1 exemplifies cell visual appearance in each of these sets. The first three rows of the table are real cell lines: CHO adherent cells (cf. Figure 4.1a), L929 adherent cells (cf. Figure 4.1b), and Sf21 suspension cells (cf. Figure 4.1c). By the term *adherent cell line*, we mean that almost all cells were adherent. Due to biological reasons, it was not possible to force all cells to adhere. Figure 4.1b exemplifies this case where some cells in the adherent cell line L929 are in suspension.

Two bioprocess engineering experts have manually labeled cells in the three real cell lines using the LabelMe annotation framework [Russ 08]. The total number of

[1] Images along with their ground-truth files are available at `http://www5.cs.fau.de/~mualla/#imagedb`

Cell line	Description	Images	Cells	Resolution
CHO	Real CHO adherent cells	6	1431	1280 × 960
L929	Real L929 adherent cells	5	1078	1280 × 960
Sf21	Real Sf21 cells in suspension	5	1001	1280 × 960
Simulated A	Simulated cells with SNR ≈ 63	100	15000	1200 × 1200
Simulated B	Simulated cells with SNR ≈ 0.07	100	15000	1200 × 1200

Table 4.1: Summary of the simulated and real standard bright field image sets

manually labeled cells is 3510. The result of this labeling is cell delineation (segmentation), so that each cell is given a unique identifier and each pixel can be thus assigned to a specific cell or to background. Cell culturing process of these cell lines is explained in Section 4.1.1 while image acquisition details are described in Section 4.1.2.

The last two rows of Table 4.1 describe the simulated images. SIMCEP [Lehm 07] was employed to simulate two cell lines. The first (cf. Figure 4.1d) is simulated with high SNR, while the second (cf. Figure 4.1e) is simulated under severe Gaussian noise conditions. Details about the simulation process are given in Section 4.1.3.

4.1.1 Cell Culturing

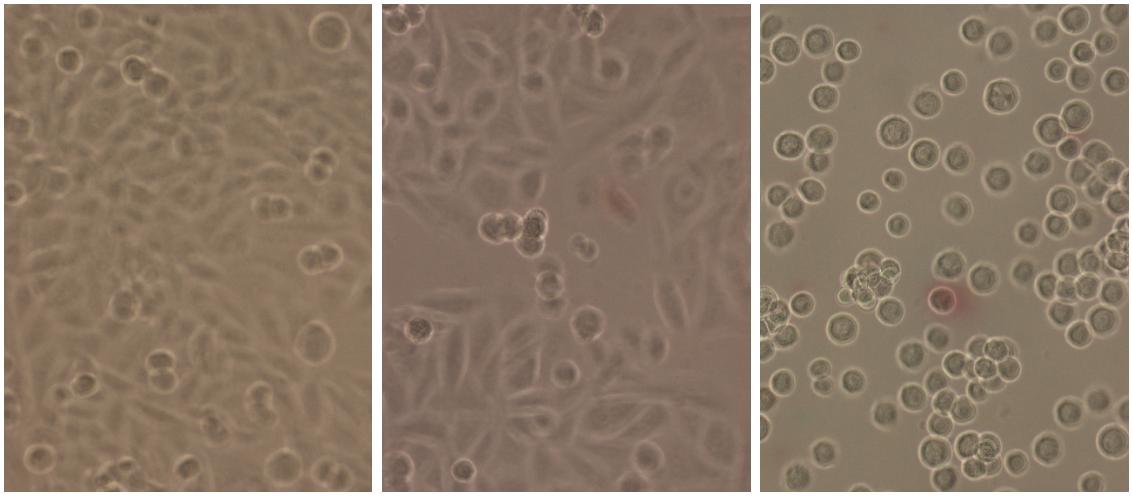
CHO-K1 epithelial-like cells and L929 murine fibroblast cells were pre-cultured and maintained in exponential growth phase in T-25 polystyrene culture flasks (Sarsted 8318.10) using DMEM/Ham's F-12 (1:1) (Invitrogen 21331-046) with 10% fetal calf serum (PAA A15-102) and 4 mM Glutamine (Sigma-Aldrich G7513-100ML) at 37°C and 7% CO_2 containing atmosphere. For image acquisition, cells were detached from the T-Flask using Accutase (Sigma-Aldrich A6964-100ML) on the day before and seeded out in 24-well plate format using a working volume of 600 μl of the mentioned medium composition. Cells were allowed to attach and spread out in the well-plate for at least 18 hours but not more than 30 hours.

Sf21 insect cells were maintained in exponential growth phase in silicon solution treated shaker flasks in Ex-Cell 420 medium (Sigma-Aldrich: 24420C) at 27°C and normal air CO_2 level. On the day of investigation, an aliquot of this culture was transferred to a 24-well plate in a final volume of 600 μl and cells were allowed to sediment before image acquisition.

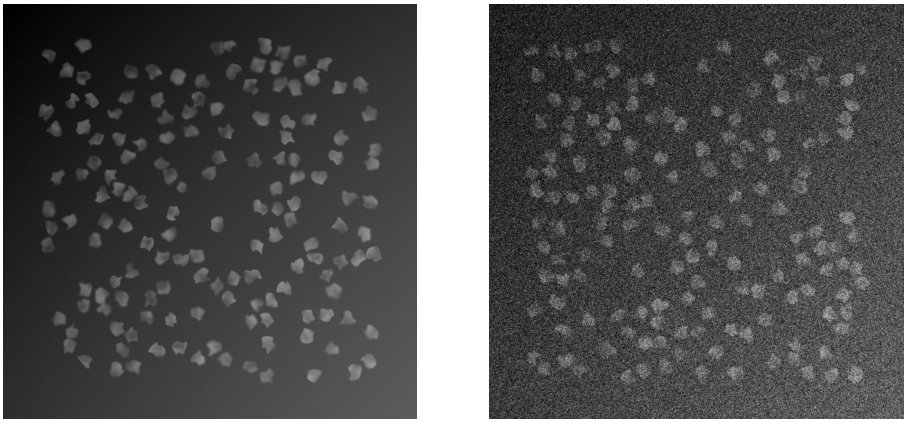
4.1.2 Real Image Acquisition

The images of the three real cell lines in Table 4.1 were manually acquired with an inverted Nikon Eclipse TE2000U microscope using Nikon's USB camera. Cells were illuminated by a halogen light bulb for standard bright field microscopy. The used microscope's objective has 20× magnification, 0.45 numerical aperture, and 7.4 mm working distance. Image resolution is 1280 × 960 pixels with 0.49 μm/pixel.

The most important acquisition parameter is probably the defocus distance. This distance was empirically set to +30 μm for the adherent cell lines and +15 μm for the suspension cell line. In addition to the previous focus level, we acquired



(a) CHO adherent (b) L929 adherent (c) Sf21 suspension



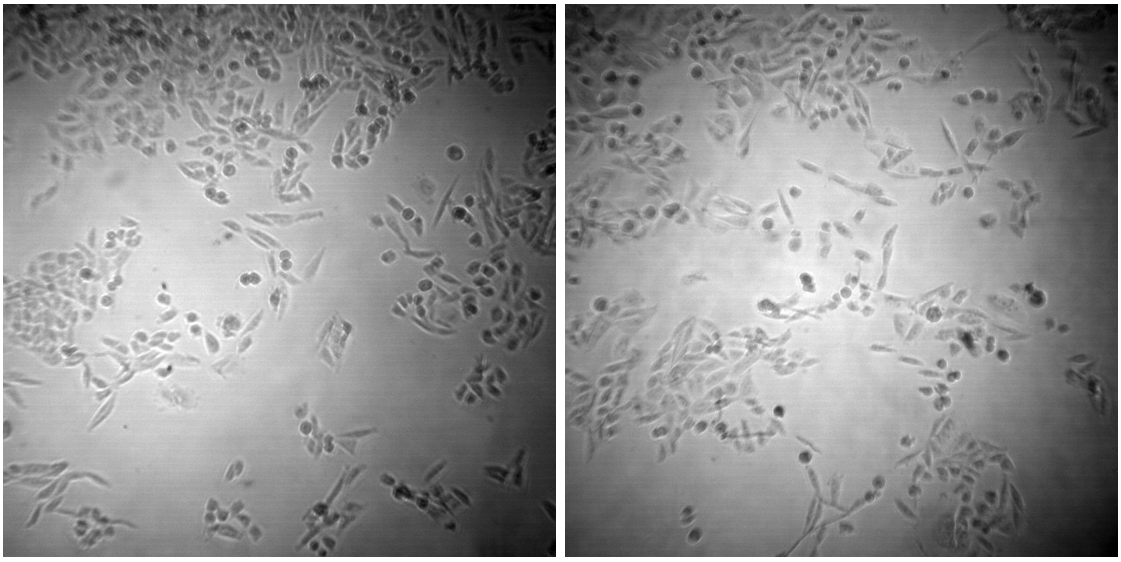
(d) Simulated A, SNR ≈ 63.0 (e) Simulated B, SNR ≈ 0.07

Figure 4.1: Demonstration of simulated and real standard bright field images: the defocus distance in (a), (b), and (c) is $+30$ μm, $+30$ μm, and $+15$ μm, respectively.

images at levels 0 and -30 μm for the adherent cell lines and at levels 0 and -15 μm for the suspension cell line. Acquiring images at multiple focus levels is needed for two reasons: 1) We compare our supervised cell detection in Chapter 5 with other approaches which require images at multiple focus levels. 2) Phase retrieval conducted in Chapter 6 requires information of intensity variation as a function of defocus distance.

4.1.3 Image Simulation

As mentioned above, the software in [Lehm 07] was utilized to simulate the two artificial cell lines in Table 4.1. It is important to point out that this software was designed for fluorescence microscopy. Nonetheless, we chose to use it for the following reasons: Firstly, to the best of our knowledge, there is no simulation software available for bright field microscopy. Secondly, if cytoplasm is excluded from the sim-



(a) A COSIR image acquired from channel A3

(b) A COSIR image acquired from channel A5

Figure 4.2: Example images acquired by the 24-channels COSIR system

ulation, cell nuclei resemble the negatively defocused bright field images even though this resemblance is partial due to differences at cell borders.

Cells were generated according to a shape model described in [Lehm 07] with dynamic range equal to 0.3 of the allowed image bins number. An illumination field with scale 10 was then added to each image. This illumination scale is defined as the ratio between illumination energy (sum of squared values) and ideal-image energy. Afterwards, white Gaussian noise was added so that the signal to noise ratio is approximately 63 in the first cell line (cf. Figure 4.1d) and 0.07 in the second (cf. Figure 4.1e). SNR is the ratio between ideal-image energy and noise energy. The illumination energy does not contribute to the SNR.

4.2 COSIR Images

During the development of COSIR hardware, a large number of images were obtained for calibrating the hardware components and verifying image quality. At the end of the financial support period of the COSIR project (end 2013), a new set of images (cf. Figure 4.2) was acquired as to reflect the most recent status of the hardware. The images were preprocessed according to the COSIR preprocessing pipeline partially described in [Scho 13]. Unlike the other image sets used in this work, COSIR images were not labeled (except a single image used for training in Chapter 5). They were thus utilized only in a qualitative evaluation of cell detection algorithms.

Cell line	Images	Cells	Ground truth	Resolution	Source
HeLa	11	1156	center dots	400×400	[Arte 12]
Bovine	10	2584	border delineation	680×512	[Pan 10]

Table 4.2: Summary of the phase contrast image sets

4.3 Phase Contrast

Besides bright field microscopy, we compared our cell detection algorithms with other approaches on phase contrast microscopy. The phase contrast image sets are listed in Table 4.2 along with their corresponding sources. As shown in the table, the ground-truth type in the dataset of [Pan 10] is cell border delineation, which is similar to ground-truth type of the standard bright field images described in Section 4.1. On the other hand, in the dataset of [Arte 12], a dot is marked at the center of each cell without delineating cell borders. As it will become clear in the next chapters, this difference in ground-truth type has implications on both training and evaluation of cell detection algorithms.

Chapter 5

Supervised Cell Detection

Considerable parts of this chapter were published in:
F. Mualla, S. Schöll, B. Sommerfeldt, A. Maier, and J. Hornegger. "Automatic cell detection in bright-field microscope images using SIFT, random forests, and hierarchical clustering". *Medical Imaging, IEEE Transactions on*, Vol. 32, No. 12, pp. 2274–2286, December 2013.

5.1 Motivation

We think that a *good* cell detection system should fulfill the following criteria:

1. It has to be automatic, i. e. no manual parameter tuning is required.
2. It should be invariant to cell size and orientation.
3. It should be invariant to illumination conditions.
4. If tracking is required after detection, then the detected cells should be equipped with trackable features.
5. The degradation of detection rate due to the aforementioned invariance requirements needs to be minimal.
6. It can learn to detect cells from a small number of labeled images.
7. It is efficient with respect to both training and detection time.

With these criteria in mind, we developed a system for unstained cell detection based on supervised learning. In this chapter, we explain this system in detail and show its robustness by conducting intensive experimental evaluations.

5.2 System Overview

A cell is expressed in bright field images as one or more *blobs* in intensity. As explained in detail in Section 3.1, a blob is an extremum of the γ-normalized LoG in scale-space.

In SIFT, it is concretely an extremum of the $\mathrm{DoG}_{\gamma 1}$ in scale-space which is again the γ-normalized difference of Gaussians for $\gamma = 1$.

The proposed supervised cell detection algorithm is roughly demonstrated in Figure 5.1. The algorithm starts by extracting SIFT keypoints from an input image. These keypoints are classified into cell and background keypoints. For each cell keypoint, an intensity profile to each *nearby* cell keypoint is extracted and classified as either *inner* or *cross*. A profile between two keypoints is termed *inner* if these two keypoints belong to the same cell. Otherwise, it is described as *cross*. The output of the profile classifier is probabilistic. The probability that a profile between two keypoints is inner can be seen as a similarity measure between the keypoints. Based on this similarity measure, an agglomerative hierarchical clustering of the keypoints with a customized linkage method is applied. A weighted mean of the keypoint coordinates inside each cluster marks then a detected cell.

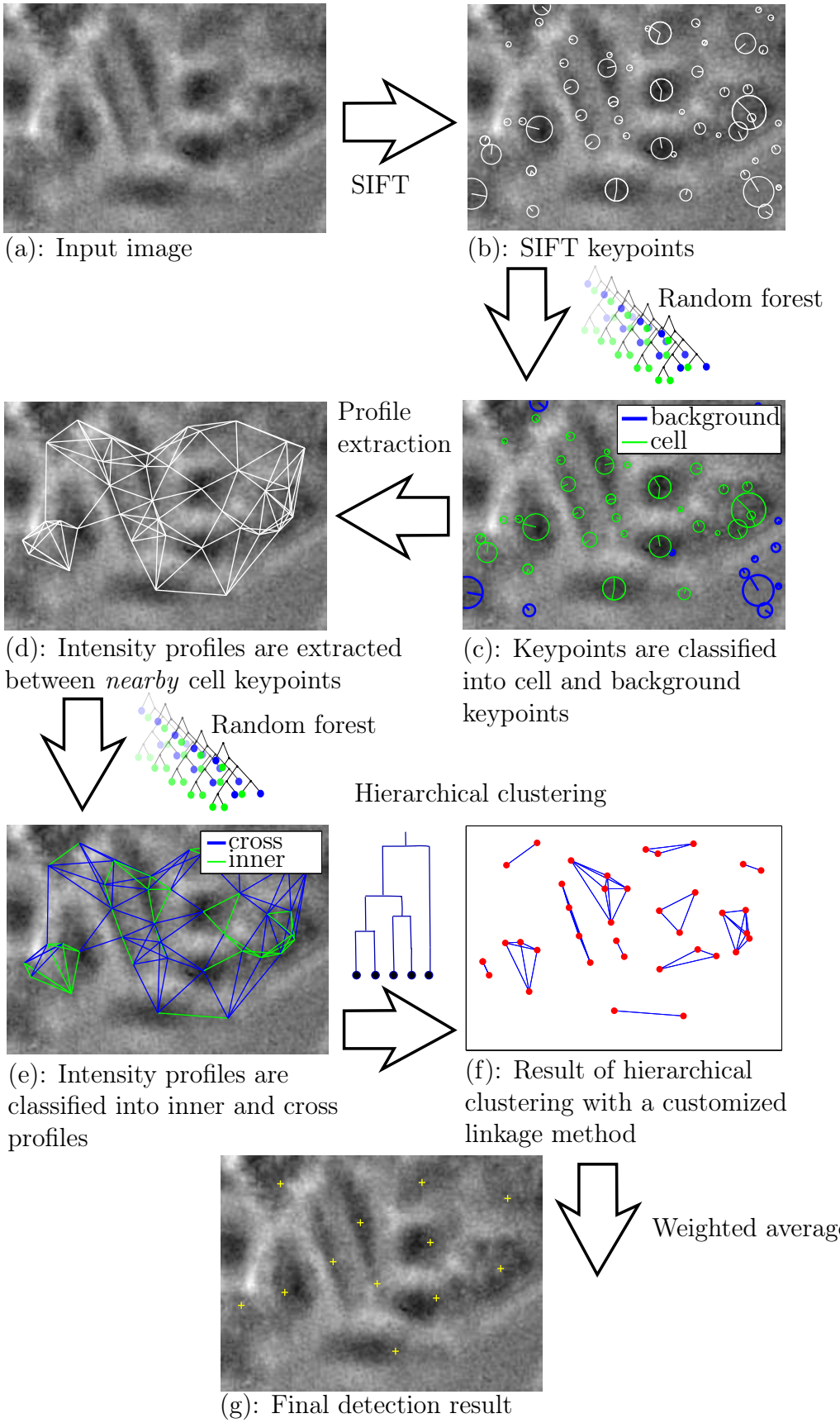
Before using these classifiers, the system must be trained. For training, a set of mask/image pairs is required. The masks should contain border delineation of cells, i. e. manual segmentation. The images should be defocused and the same defocus distance needs to be used for training and detection (testing). Defocusing can be either in the positive or the negative direction. It is important for at least two reasons: Firstly, as mentioned in previous chapters (see Section 1.1 and Section 2.6), some adherent cells are totally invisible at focus. Secondly, defocusing smoothes out tiny details which may degrade system performance in terms of time and detection rate.

No manual tuning of any parameter is required neither in training nor in detection. The system learns its parameters automatically in a scale- and orientation-invariant manner.

The rest of this chapter is organized as follows: Section 5.3 describes the features and classifier model of the keypoint classifier. Section 5.4 discusses different aspects of the profile learning. In Section 5.5, the hierarchical clustering step is explained and the customized linkage method is introduced. Section 5.6 provides details of the training phase. Measures of detection quality are defined in Section 5.7. Section 5.8 contains experimental evaluation. The chapter ends with a detailed discussion in Section 5.9.

5.3 Keypoint Learning

SIFT keypoints are extracted by searching for $\mathrm{DoG}_{\gamma 1}$ extrema at all possible scales. In other words, the number of octaves $\mathfrak{O}$ is approximately $log_2\left(\min\left(N_{\mathrm{Rows}}, N_{\mathrm{Columns}}\right)\right)$, where N_{Rows} and N_{Columns} are the number of image's rows and columns, respectively. The number of scales per octave $\mathfrak{S}$ was set to 3. As mentioned in Section 3.1.4, this value was shown to yield stable keypoints. SIFT keypoints can be thresholded according to their strength represented by the $\mathrm{DoG}_{\gamma 1}$ value and/or their elongation given by the PCR value (cf. Section 3.1.4). As we aim to design an automatic pipeline, we consider all SIFT keypoints without setting any manual threshold. Nonetheless, later on, we use training data to learn a relevant $\mathrm{DoG}_{\gamma 1}$ threshold.



(a): Input image

(b): SIFT keypoints

(c): Keypoints are classified into cell and background keypoints

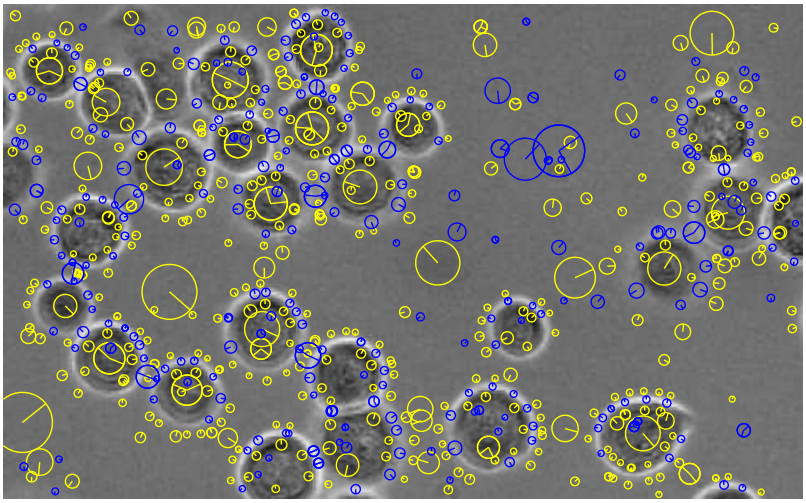
(d): Intensity profiles are extracted between *nearby* cell keypoints

(e): Intensity profiles are classified into inner and cross profiles

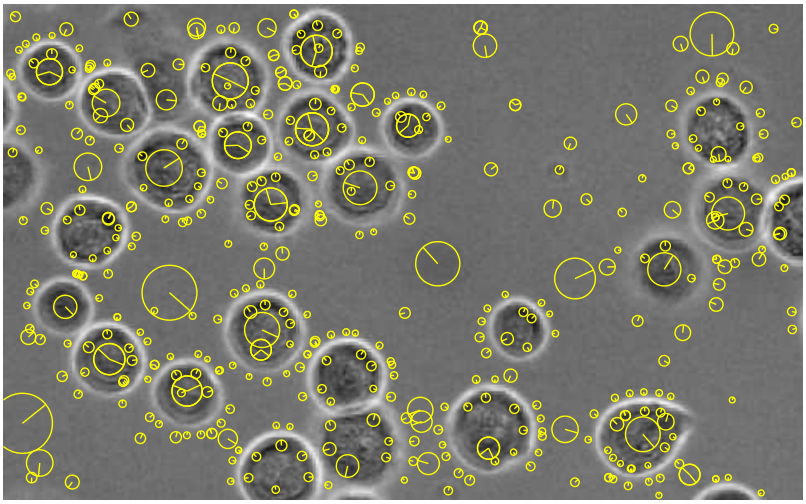
(f): Result of hierarchical clustering with a customized linkage method

(g): Final detection result

Figure 5.1: A rough overview of the proposed supervised cell detection pipeline



(a) Result of SIFT keypoint extraction: yellow indicates keypoints with $\mathrm{DoG}_{\gamma 1} > 0$ (dark appearance) while blue indicates keypoints with $\mathrm{DoG}_{\gamma 1} < 0$ (bright appearance).



(b) Dominant blob type is determined by Eq. 5.1 to be the keypoints with positive $\mathrm{DoG}_{\gamma 1}$. Keypoints with negative $\mathrm{DoG}_{\gamma 1}$ are thus discarded.

Figure 5.2: Illustration of automatic determination of the dominant blob type

In order to determine, whether cells in the training data tend to have a bright or dark appearance, we perform a calibration step before the actual cell detection procedure. An example of dark appearance can be seen in figures 4.1a, 4.1b, and 4.1c while bright appearance is exemplified in figures 4.1d and 4.1e. In this thesis, we call a set of keypoints *one-sided* if all the keypoints in the set have positive $\mathrm{DoG}_{\gamma 1}$ values or all of them have negative $\mathrm{DoG}_{\gamma 1}$ values. In order to automatically detect the keypoint type which fits the training data, we introduce the following measure:

$$\mathrm{BT} = \mathrm{sign}\left(\frac{\sum_{j=1}^{N_{ckp}} \sigma(\mathbf{p}_j) \left|\mathrm{DoG}_{\gamma 1}\left(\mathbf{p}_j\right)\right| \mathrm{HS}\left(\mathrm{DoG}_{\gamma 1}\left(\mathbf{p}_j\right)\right)}{\sum_{j=1}^{N_{ckp}} \sigma\left(\mathbf{p}_j\right) \left|\mathrm{DoG}_{\gamma 1}\left(\mathbf{p}_j\right)\right|} - \frac{1}{2} \right), \tag{5.1}$$

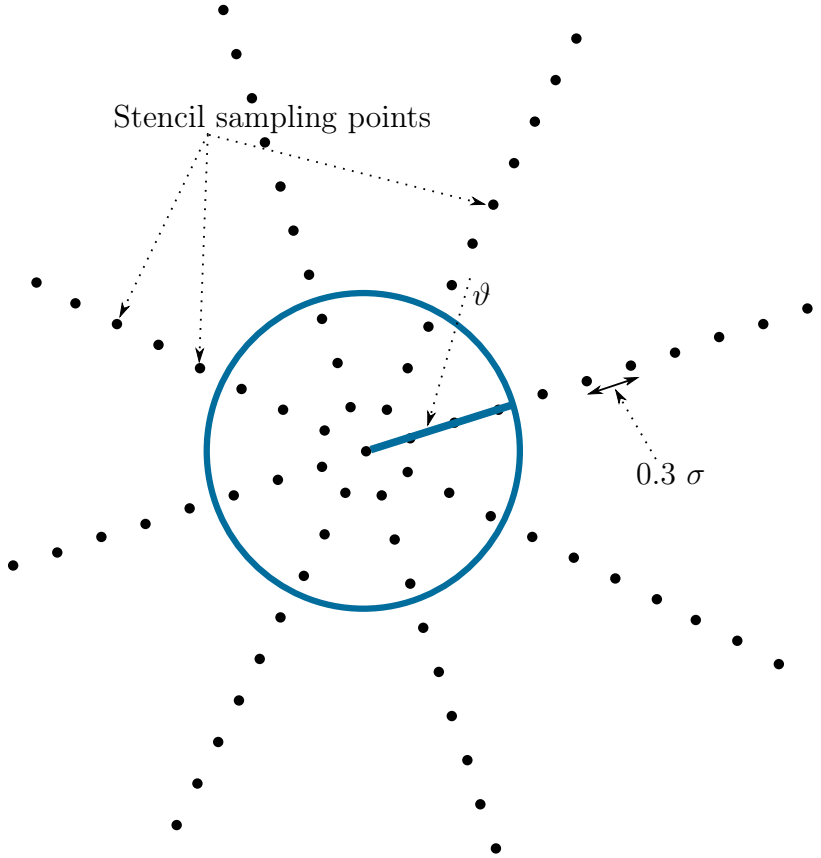


Figure 5.3: Illustration of the keypoint-based radial intensity stencil obtained by adjusting fixed-size intensity stencils to SIFT keypoints. The stencil is aligned with the keypoint orientation ϑ and the distance between two successive nodes is set to 0.3 σ, where σ is the keypoint scale.

where $\mathbf{p}_j$, $j = 1 \,..\, N_{ckp}$ are the cell keypoints in the training data (background keypoints are not used), N_{ckp} is their number, σ is the keypoint scale, sign is the sign function (returns the sign of its operand), and HS is the Heaviside step function defined as follows:

$$\begin{aligned} \mathrm{HS}(\varrho) &= 1 \text{ when } \varrho > 0 \\ &= 0 \text{ otherwise.} \end{aligned} \tag{5.2}$$

If BT evaluates to +1, one can conclude that the dominant blob type is the type defined by $\mathrm{DoG}_{\gamma 1} > 0$ which describes the black-on-white blobs. The system thus ignores keypoints with negative $\mathrm{DoG}_{\gamma 1}$ values during training and detection. This case is illustrated in Figure 5.2. On the other hand, when BT = −1, the system decides for white-on-black keypoints and ignores keypoints with positive $\mathrm{DoG}_{\gamma 1}$ values.

After that, the maximum $|\mathrm{DoG}_{\gamma 1}|$ in each training cell is computed and the first percentile of all resulting maxima is considered a SIFT threshold. During training and detection, SIFT keypoints which have lower $|\mathrm{DoG}_{\gamma 1}|$ than this threshold will be discarded. The goal of this step is to eliminate instable keypoints.

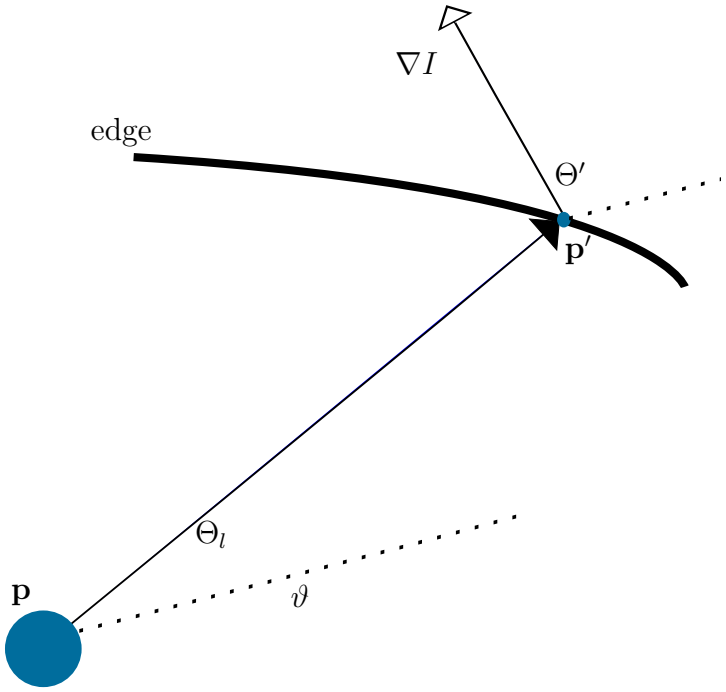


Figure 5.4: Illustration of the keypoint-based ray features obtained by adjusting standard ray features to SIFT keypoints. The figure is an adapted version of the pixel-based ray features in [Smit 09]. Please refer to text for explanation.

5.3.1 Keypoint Features

SIFT keypoints in cell images result from cell structures, but also from debris, noise in the background, and other image artifacts. It is possible to eliminate some irrelevant keypoints by imposing a high $\text{DoG}_{\gamma 1}$ threshold which excludes weak structures or by imposing a low PCR threshold which excludes elongated structures (cf. Section 3.1.4). However, some cell keypoints may have high principal curvatures ratio because of cell elongation. Other cell keypoints may have low DoG value due to cell adherence or insufficient contrast of the considered image modality. Consequently, it is not always possible to separate cell keypoints from non-cell keypoints reliably using these two features. In order to achieve this separation in a reliable manner, we used several sets of features from the literature and adjusted them. We utilized SIFT to make these features scale- and orientation-invariant. Moreover, we made them invariant to *local shift* of intensity. This local shift is simply a constant addition to the image intensity in a small (and hence local) region. The importance of invariance to the local shift of intensity can be clarified as follows: under the assumption that the illumination field in a small region of the image can be approximated by a constant, invariance to local intensity shifts contributes to illumination invariance.

At each keypoint, the following feature sets are extracted:

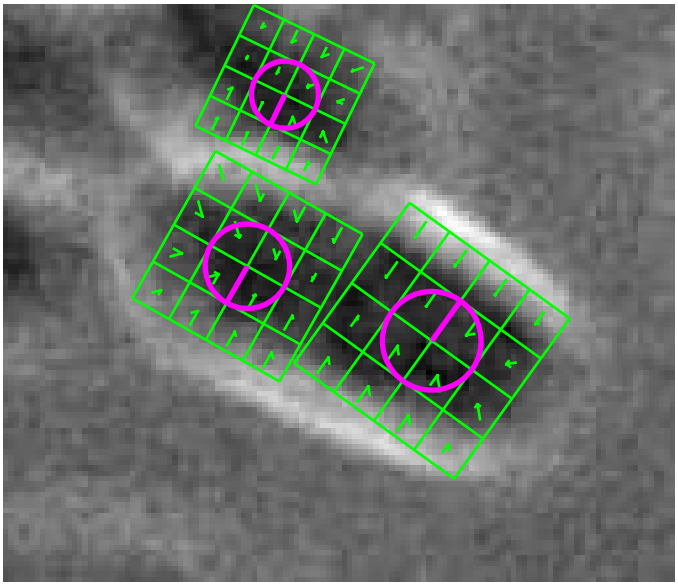


Figure 5.5: Demonstration of SIFT descriptors with SIFT magnification factor $M = 1$. Each descriptor is composed of 16 subregions with a histogram of gradient orientations of 8 bins in each of them. For clarity of the figure, only a subset of descriptors are shown.

1. Intensity stencil: we computed intensity stencils [Jurr 10, Mitt 10] around each keypoint. As shown in Figure 5.3, we align the stencil with the keypoint orientation and measure the distance between sampling points in units of keypoint scale σ instead of pixels. The result is a scale- and orientation-invariant stencil. In order to make the stencil invariant to local shift of intensity, we subtract the mean intensity of the stencil from all stencil nodes. The intensity values at the stencil nodes after the aforementioned subtraction form the stencil feature set.

2. Ray features: in order to compute ray features [Smit 09] at a keypoint $\mathbf{p}$ (cf. Figure 5.4) in an image I, the closest edge point $\mathbf{p}'$ along a direction Θ_l is found. We discretize Θ_l in 8 values, i. e. $l = 1..8$. For each keypoint $\mathbf{p}$ and direction Θ_l, we extract the following features [Smit 09]:

   - the distance $\mathrm{Ray}_d(\mathbf{p}, \Theta_l)$ between $\mathbf{p}$ and $\mathbf{p}'$.
   - the gradient norm $\mathrm{Ray}_n(\mathbf{p}, \Theta_l)$ at $\mathbf{p}'$.
   - the gradient angle at $\mathbf{p}'$, i. e. $\mathrm{Ray}_a(\mathbf{p}, \Theta_l) = \Theta'$.
   - the distance difference along two different directions Θ_l and $\Theta_{l'}$: $\mathrm{Ray}_{dd}(\mathbf{p}, \Theta_l, \Theta_{l'}) = |\mathrm{Ray}_d(\mathbf{p}, \Theta_l) - \mathrm{Ray}_d(\mathbf{p}, \Theta_{l'})|$.

   Since eight values of the angle Θ are used, each of the first three items yields 8 features per keypoint. The fourth item involves each combination of two different directions. It thus yields $\frac{8 \cdot 7}{2} = 28$ features. In total, we obtain $8 \cdot 3 + 28 = 52$ ray features.

   Ray features are well-designed but are sensitive to scale and orientation. In order to make them orientation-invariant, we define all angles, i. e. the eight

Θ_l angles and the gradient angle feature $\mathrm{Ray}_a(\mathbf{p}, \Theta_l)$, with respect to the keypoint orientation. In order to make ray features scale-invariant, we measure the distances $\mathrm{Ray}_d(\mathbf{p}, \Theta_l)$ and $\mathrm{Ray}_{dd}(\mathbf{p}, \Theta_l, \Theta_{l'})$ in terms of keypoint scale σ. Furthermore, in order to make the gradient norm $\mathrm{Ray}_n(\mathbf{p}, \Theta_l)$ scale-invariant, we compute the gradient using the following equation for its x component:

$$\frac{\partial I(\mathbf{p}')}{\partial x} = I(p'_x + \upsilon\sigma(\mathbf{p}), p'_y) - I(p'_x, p'_y), \tag{5.3}$$

where $\sigma(\mathbf{p})$ is the scale of the keypoint at $\mathbf{p}$. υ is a constant that we set to 1. A similar equation is used for the y component. Before applying Eq. 5.3, the image is smoothed using a Gaussian kernel with a standard deviation equal to 1. The edges for computing ray features are obtained using Canny edge detection [Cann 86]. The thresholds were set to the default values in the Matlab implementation of Canny.

3. Variance map: based on the variance map [Wu 95], we created a keypoint-based scale-invariant version by taking a variable-size neighborhood with a size proportional to the keypoint scale. For each keypoint $\mathbf{p}$, we extract three variance map features $\mathrm{VMap}(\mathbf{p}, 2)$, $\mathrm{VMap}(\mathbf{p}, 4)$, and $\mathrm{VMap}(\mathbf{p}, 6)$ which correspond to variance map in a square neighborhood of side length $2\sigma(\mathbf{p})$, $4\sigma(\mathbf{p})$, and $6\sigma(\mathbf{p})$, respectively. The map is by construction invariant to the local shift of intensity as adding a constant to a set of values does not change the variance of this set.

4. SIFT descriptors: SIFT features were used according to the original publication [Lowe 99], as they are inherently scale- and orientation-invariant and partially illumination-invariant. We set the magnification factor M to 1 (cf. Section 3.1.5) so that the dimensions of the SIFT descriptor are $4\sigma \times 4\sigma$. With this setting, the descriptor area is not large and it is thus unlikely to span too many cells. Figure 5.5 demonstrates descriptor features on cell images.

5. Other features: the values of the $\mathrm{DoG}_{\gamma 1}$ and the principal curvatures ratio PCR at each keypoint were also obtained from SIFT. As the $\mathrm{DoG}_{\gamma 1}$ value is a γ-normalized Gaussian derivative with $\gamma = 1$ (cf. Section 3.1.3), it can be assumed scale-invariant. It is also rotation-invariant because the intensity Laplacian, of which the $\mathrm{DoG}_{\gamma 1}$ is an approximation, is the sum of the two eigenvalues of the Hessian of image intensity. These eigenvalues are known to be rotation-invariant. On the other hand, the PCR is dependent on the eigenvalues ratio of the Hessian of $\mathrm{DoG}_{\gamma 1}$. Therefore, it can be also assumed scale- and orientation-invariant.

5.3.2 Keypoint Classifier

We chose the random forest [Brei 01] as a background/cell classifier. One motivation for this selection is that random forests can be used without parameter tuning. In addition, as mentioned in Section 3.2.2, they are robust against overfitting and practical in terms of training time. After [Khos 07], we set the number of trees N_{Tr} to 500 and the number of randomly selected features at each node N_{Rand} to $N_F/5$, where

N_F is the number of features. Since it is not guaranteed that the two classes *cell* and *background* are balanced, we chose to use a balanced random forest [Chen 04]. As also mentioned in Section 3.2.2, the size of bootstrap replicate is set in this version of random forest to the cardinality of minority class.

5.4 Profile Learning

The keypoints which were classified as background by the keypoint classifier are discarded. The remaining ones are thus cell keypoints. The number of keypoints inside each cell depends on noise level, cellular details level, defocus distance, cell shape, and SIFT parameters (e. g. the number of scales per octave $\mathfrak{S}$). In a nutshell, the goal of profile learning is to connect keypoints which belong to the same cell together so that they are recognized as one cell.

In order to decide whether two cell keypoints belong to the same cell, we extract an intensity profile $\mathrm{prof}(\chi)$ between them, where $\chi = 1 \,..\, N_{\mathrm{PPoints}}$ are the sampling points along the profile, and N_{PPoints} is the number of profile points. We then extract profile features and use them to classify the profile as either *inner* or *cross*.

5.4.1 Profile Features and Classifier

The following features are extracted for each intensity profile $\mathrm{prof}(\chi)$:

- Standard deviation, skewness, and kurtosis of the profile.
- Standard deviation, skewness, kurtosis, maximum, minimum, and mean of first derivative $\frac{d\mathrm{prof}}{d\chi}$ and second derivative $\frac{d^2\mathrm{prof}}{d\chi^2}$ of the profile.
- Two other features:

$$V_1 = \max(\mathrm{prof}) - \min(\mathrm{prof}) \tag{5.4}$$

$$V_2 = \mathrm{prof}(1) - 2\max(\mathrm{prof}) + \mathrm{prof}(N_{\mathrm{PPoints}}) \tag{5.5}$$

The profiles are sampled using a fixed number of points $N_{\mathrm{PPoints}} = 50$. The derivatives are Gaussian derivatives with $\sigma_{\mathrm{Profile}} = 0.1\, N_{\mathrm{PPoints}}$. As the scale of derivative, i. e. the previous $\sigma_{\mathrm{Profile}}$, is measured in units of sampling points, not in pixels, the derivative signal is to a large extent scale-invariant. Note that all profile features are also invariant to the local shift of intensity and that the mean, maximum, and minimum of the profile do not belong to the profile feature set because they are sensitive to this shift.

As a classifier model, we employ a balanced random forest based on the same justifications and using the same parameters mentioned in Section 5.3.2.

5.4.2 Learning to Extract Small-length Profiles

For the training of the profile classifier and during the detection (testing) phase, the algorithm extracts a profile between two keypoints only if they are *nearby*. This makes sense for three reasons:

1. In Section 5.3, it was mentioned that the keypoint features were made invariant to local shift of intensity in order to gain robustness against illumination conditions. The assumption was that, in a local region, the illumination can be assumed to be constant. We want to apply a similar principle for profile features. For this reason, it is desired to have intensity profiles of short length so that illumination can be approximated by a constant along the profile.

2. The goal of extracting intensity profiles is to rank keypoints whether they belong to the same cell or to different cells. This is, however, necessary only for keypoints which belong to nearby cells.

3. In terms of runtime, it is more efficient to extract intensity profiles between nearby keypoints instead of doing that for each keypoint pair in the considered image.

In order to achieve this goal in a scale-invariant manner, the algorithm allows for learning the *maximum inner profile length* from training data. Naturally, it is inappropriate to measure the profile length in pixels. We use the average scale $\mathrm{LU}(I)$ of all *cell* keypoints inside an image I as a profile **l**ength **u**nit for this image. The unit $\mathrm{LU}(I)$ can be computed during training because cell keypoints are known from ground truth. However, it can be also computed during testing (ground truth is not available) since cell keypoints are obtainable as outcome of the cell/background classification step.

In general, training images may have a scale which is different from the scale of testing images. Additionally, training images themselves may have different scales. Therefore, the unit $\mathrm{LU}(I)$ is computed independently for each training/testing image.

During training, the maximum inner profile length MaxL_j in each training image I_j is computed in terms of $\mathrm{LU}(I_j)$. Then the maximum of all the MaxL_j values is considered the maximum inner profile length in the training data:

$$\mathrm{MaxL} = \max(\mathrm{MaxL}_j), \quad j = 1 \,..\, N_{\mathrm{Training}},$$

where N_{Training} is the number of training images. Unlike LU, MaxL is a scale-invariant measure. It is saved as output of the training and used during testing to decide whether two keypoints are nearby. It is, however, needed also during training, as only short-length profiles may be used for training the profile classifier.

More specifically, given a cell keypoint $\mathbf{p}_1$ in a training or testing image I, another cell keypoint $\mathbf{p}_2$ in I is considered *nearby* if the Euclidean distance between the two keypoints, in units of $\mathrm{LU}(I)$, is smaller than $\zeta\mathrm{MaxL}$. The symbol ζ denotes a safety parameter that we set to 2.

5.4.3 Profile Expansion

The image area which is sampled by a single intensity profile is actually very small. It is thus plausible to expect an improvement in detection accuracy when a profile captures information from a wider image area. Instead of extracting one profile between the two considered keypoints, one could extract a set of profiles, i. e. several parallel profiles as demonstrated in Figure 5.6. The geometry of the set can be

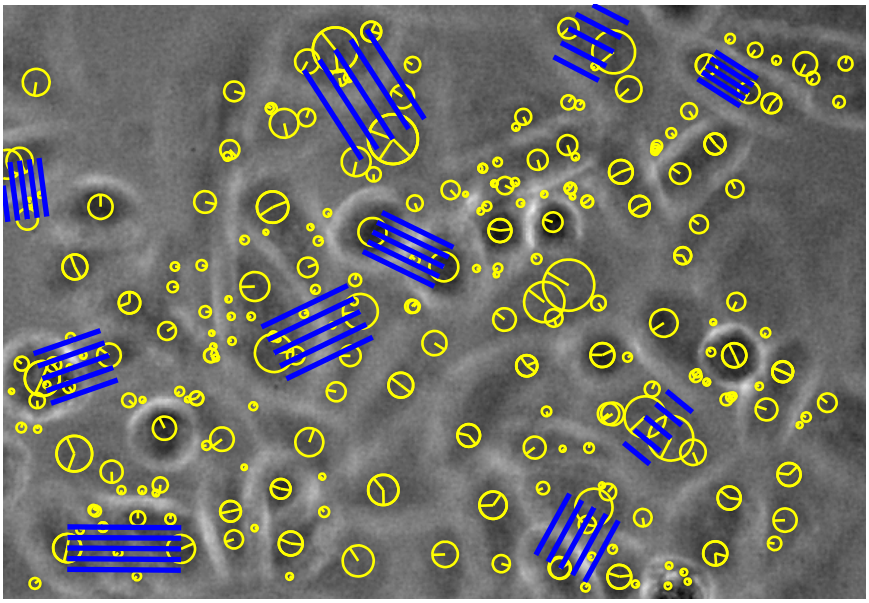


Figure 5.6: Profile sets between different keypoints: only a subset of the profile sets was drawn in order to preserve figure clarity.

described, for instance, in terms of maximum scale of the two keypoints. In Figure 5.6, a profile set between two keypoints $\mathbf{p}_l$ and $\mathbf{p}_q$ contains five parallel profiles, one profile every $0.75 \cdot \sigma_{lq}^{\max}$. The symbol $\sigma_{lq}^{\max}$ denotes the maximum scale of the two considered keypoints $\mathbf{p}_l$ and $\mathbf{p}_q$. This specific setup of the profile set is also the one which is used later in our evaluation of cell detection performance under the use of profile sets.

Alternatively, smoothing is a well-known fast and simple means of information consolidation. The natural question which arises here is about the smoothing scale. We make smoothing *scale adaptive* by setting the standard deviation of the Gaussian kernel which is used to smooth an image I to the mean scale of the one-sided (cf. Section 5.3) keypoints $\bar{\sigma}$ of I. We call this process scale adaptive smoothing (SAS). A third approach is to combine the previous two ones. We use the SAS method for our cell detection pipeline. The reasons behind this selection will become clear in the evaluation section.

5.4.4 Handling Lack of Cross/Inner Profiles for Training

If the number of inner profiles in the training data is too small, the algorithm extracts artificial inner profiles until the number of inner profiles in the training data is at least N_{Inner}. This is done by sampling N_{Inner} inner intensity profiles even when there is only one keypoint per cell. In this case, the extracted inner profile is started at a cell keypoint from one side and terminated at cell border from the other side.

If the number of cross profiles in the training data is too small, the algorithm generates artificial cross profiles until the number of cross profiles in the training data is at least N_{Cross}. They are generated by shifting the considered training image I by $\varsigma \text{LU}(I)$ and then overlaying the original and the shifted version. The parameter ς was set to 6. Both N_{Inner} and N_{Cross} were set to 15 in our experiments.

5.5 Hierarchical Clustering

In this section, we address the problem of combining the results of keypoint learning and profile learning in order to detect cells. One could employ a graph-based approach: assume a graph GH with cell keypoints as nodes. Two nodes are connected if the profile between them is an inner profile. The nodes of GH are obtained from keypoint classification while the edges are obtained from profile classification. Intuitively, each connected component in GH can be seen as a detected cell. This technique will be referred to later in this chapter as the connected components (CC).

Utilization of the available information using CC is suboptimal. Alternatively, one can think of clustering the keypoints in an agglomerative manner starting from the most reliable ones: the agglomerative hierarchical clustering of the keypoints. As mentioned in Section 3.3, AHC is characterized by a similarity measure and a linkage method. Two cell keypoints $\mathbf{p}_j$ and $\mathbf{p}_l$ are similar if they belong to the same cell. Thus, it is plausible to define the similarity between $\mathbf{p}_j$ and $\mathbf{p}_l$ as the probability that the profile between them is an inner profile $\widehat{P}_{inner}(\mathbf{p}_j, \mathbf{p}_l)$. This is equivalent to defining $\widehat{P}_{cross}(\mathbf{p}_j, \mathbf{p}_l) = 1 - \widehat{P}_{inner}(\mathbf{p}_j, \mathbf{p}_l)$ as a dissimilarity measure between the two cell keypoints.

5.5.1 Customized Linkage Method

One plausible choice for the linkage method is the group average link (cf. Eq. 3.23). In order to incorporate more information, we use a customized group average linkage instead of the traditional one. According to this customized linkage, the dissimilarity between two clusters $\mathbb{A}$ and $\mathbb{B}$ is:

$$\Pi_{\text{Customized}}(\mathbb{A}, \mathbb{B}) = \sum_{j=1}^{|\mathbb{A}|} \sum_{l=1}^{|\mathbb{B}|} \frac{\xi_{jl}}{\Xi_{\mathbb{A},\mathbb{B}}} \pi(\mathbf{p}_j, \mathbf{p}_l) \tag{5.6}$$

$$\pi(\mathbf{p}_j, \mathbf{p}_l) = \widehat{P}_{cross}(\mathbf{p}_j, \mathbf{p}_l) \tag{5.7}$$

$$\xi_{jl} = \frac{|\text{DoG}_{\gamma 1}(\mathbf{p}_j)|\sigma(\mathbf{p}_j)|\text{DoG}_{\gamma 1}(\mathbf{p}_l)|\sigma(\mathbf{p}_l)}{\|\mathbf{p}_j - \mathbf{p}_l\|^2} \tag{5.8}$$

$$\Xi_{\mathbb{A},\mathbb{B}} = \sum_{j=1}^{|\mathbb{A}|} \sum_{l=1}^{|\mathbb{B}|} \xi_{jl}, \tag{5.9}$$

where $\mathbf{p}_j$ is a keypoint in the cluster $\mathbb{A}$. $\mathbf{p}_l$ is a keypoint in the cluster $\mathbb{B}$. $|\mathbb{A}|$ and $|\mathbb{B}|$ are the cardinalities of $\mathbb{A}$ and $\mathbb{B}$, respectively. ξ_{jl} is the weight of the dissimilarity between $\mathbf{p}_j$ and $\mathbf{p}_l$. This weight is proportional to the scale and the absolute $\text{DoG}_{\gamma 1}$ of the two points and inversely proportional to the squared Euclidean norm of the profile $\|\mathbf{p}_j - \mathbf{p}_l\|^2$. Indeed, ξ_{jl} is scale-invariant because the scale and the distance terms in the numerator and the denominator have the same order. $\Xi_{\mathbb{A},\mathbb{B}}$ is the normalization factor of the weights.

The obtained dissimilarity from this customized linkage is always in the range $[0, 1]$. It can be interpreted as probability since it is a weighted sum of probabilities

where the weights also sum up to 1. This clear interpretation makes it easy to select a meaningful threshold for cutting the dendrogram: the final clusters are obtained by cutting the dendrogram at a cutoff equal to 0.5.

5.5.1.1 Lance-Williams Coefficients

As mentioned in Section 3.3.1, in the combinatorial clustering strategies, the new dissimilarities can be computed from the old ones using the Lance-Williams dissimilarity update formula [Lanc 66, Lanc 67] which we repeat here for convenience of the reader: if $\mathbb{A}$ and $\mathbb{B}$ are merged into one cluster $\mathbb{AB}$, then the dissimilarity between any other cluster $\mathbb{C}$ and the cluster $\mathbb{AB}$ is given by:

$$\Pi(\mathbb{AB},\mathbb{C}) = \beta_1\Pi(\mathbb{A},\mathbb{C}) + \beta_2\Pi(\mathbb{B},\mathbb{C}) + \beta_3\Pi(\mathbb{A},\mathbb{B}) + \beta_4|\Pi(\mathbb{A},\mathbb{C}) - \Pi(\mathbb{B},\mathbb{C})|, \quad (5.10)$$

where β_1, β_2, β_3, and β_4 are the coefficients of this linear model. The values of these coefficients for some standard linkage methods can be checked in Table 3.1. As we use a customized linkage, we need to find the corresponding coefficient values. It can be shown that the model coefficients for our customized linkage are:

$$\beta_1^{\text{Customized}} = \frac{\Xi_{\mathbb{A},\mathbb{C}}}{\Xi_{\mathbb{A},\mathbb{C}} + \Xi_{\mathbb{B},\mathbb{C}}} \quad (5.11)$$

$$\beta_2^{\text{Customized}} = \frac{\Xi_{\mathbb{B},\mathbb{C}}}{\Xi_{\mathbb{A},\mathbb{C}} + \Xi_{\mathbb{B},\mathbb{C}}} \quad (5.12)$$

$$\beta_3^{\text{Customized}} = \beta_4^{\text{Customized}} = 0 \quad (5.13)$$

Proof: By using Eq. 5.6, Eq. 5.11, Eq. 5.12, and Eq. 5.13 in the right-hand side of Eq. 5.10, we obtain LWR (Lance-Williams right-hand side):

$$\begin{aligned}
\text{LWR} &= \frac{\Xi_{\mathbb{A},\mathbb{C}}}{\Xi_{\mathbb{A},\mathbb{C}} + \Xi_{\mathbb{B},\mathbb{C}}} \sum_{j=1}^{|\mathbb{A}|}\sum_{q=1}^{|\mathbb{C}|} \frac{\xi_{jq}}{\Xi_{\mathbb{A},\mathbb{C}}}\pi(\mathbf{p}_j,\mathbf{p}_q) + \frac{\Xi_{\mathbb{B},\mathbb{C}}}{\Xi_{\mathbb{A},\mathbb{C}} + \Xi_{\mathbb{B},\mathbb{C}}} \sum_{l=1}^{|\mathbb{B}|}\sum_{q=1}^{|\mathbb{C}|} \frac{\xi_{lq}}{\Xi_{\mathbb{B},\mathbb{C}}}\pi(\mathbf{p}_l,\mathbf{p}_q) \\
&= \frac{1}{\Xi_{\mathbb{A},\mathbb{C}} + \Xi_{\mathbb{B},\mathbb{C}}} \sum_{j=1}^{|\mathbb{A}|}\sum_{q=1}^{|\mathbb{C}|} \xi_{jq}\pi(\mathbf{p}_j,\mathbf{p}_q) + \frac{1}{\Xi_{\mathbb{A},\mathbb{C}} + \Xi_{\mathbb{B},\mathbb{C}}} \sum_{l=1}^{|\mathbb{B}|}\sum_{q=1}^{|\mathbb{C}|} \xi_{lq}\pi(\mathbf{p}_l,\mathbf{p}_q) \\
&= \frac{1}{\Xi_{\mathbb{A},\mathbb{C}} + \Xi_{\mathbb{B},\mathbb{C}}} \left(\sum_{j=1}^{|\mathbb{A}|}\sum_{q=1}^{|\mathbb{C}|} \xi_{jq}\pi(\mathbf{p}_j,\mathbf{p}_q) + \sum_{l=1}^{|\mathbb{B}|}\sum_{q=1}^{|\mathbb{C}|} \xi_{lq}\pi(\mathbf{p}_l,\mathbf{p}_q) \right) \\
&= \frac{1}{\Xi_{\mathbb{A},\mathbb{C}} + \Xi_{\mathbb{B},\mathbb{C}}} \sum_{q=1}^{|\mathbb{C}|} \left(\sum_{j=1}^{|\mathbb{A}|} \xi_{qj}\pi(\mathbf{p}_q,\mathbf{p}_j) + \sum_{l=1}^{|\mathbb{B}|} \xi_{ql}\pi(\mathbf{p}_q,\mathbf{p}_l) \right). \quad (5.14)
\end{aligned}$$

In Eq. 5.14, for each element of $\mathbb{C}$, there is inside the brackets an iteration over all elements of $\mathbb{A}$ and $\mathbb{B}$. Notice, however, that the term $\xi_{qj}\pi(\mathbf{p}_q,\mathbf{p}_j)$ depends on the selection of the two keypoints $\mathbf{p}_q$ and $\mathbf{p}_j$, but not on the clusters to which these keypoints are assigned. Likewise, this applies to the term $\xi_{ql}\pi(\mathbf{p}_q,\mathbf{p}_l)$ as well. The

aforementioned iteration over the elements of $\mathbb{A}$ and $\mathbb{B}$ is thus equivalent to an iteration over the elements of their union $\mathbb{AB}$. Consequently, Eq. 5.14 can be rewritten as follows:

$$\text{LWR} = \frac{1}{\Xi_{\mathbb{A},\mathbb{C}} + \Xi_{\mathbb{B},\mathbb{C}}} \sum_{q=1}^{|\mathbb{C}|} \sum_{q'=1}^{|\mathbb{AB}|} \xi_{qq'} \pi(\mathbf{p}_q, \mathbf{p}_{q'}) \cdot \tag{5.15}$$

Moreover, based on Eq. 5.9, we can write:

$$\begin{aligned} \Xi_{\mathbb{A},\mathbb{C}} + \Xi_{\mathbb{B},\mathbb{C}} &= \sum_{q=1}^{|\mathbb{C}|} \sum_{j=1}^{|\mathbb{A}|} \xi_{qj} + \sum_{q=1}^{|\mathbb{C}|} \sum_{l=1}^{|\mathbb{B}|} \xi_{ql} \\ &= \sum_{q=1}^{|\mathbb{C}|} \left(\sum_{j=1}^{|\mathbb{A}|} \xi_{qj} + \sum_{l=1}^{|\mathbb{B}|} \xi_{ql} \right) \\ &= \sum_{q=1}^{|\mathbb{C}|} \sum_{q'=1}^{|\mathbb{AB}|} \xi_{qq'} \\ &= \Xi_{\mathbb{C},\mathbb{AB}} \cdot \end{aligned} \tag{5.16}$$

From Eq. 5.16 and Eq. 5.15, one can conclude directly that using the coefficients given by Eq. 5.11, Eq. 5.12, and Eq. 5.13 in the right-hand side of the Lance-Williams model yields the customized linkage distance between $\mathbb{AB}$ and $\mathbb{C}$ which is the left-hand side of the model.

5.5.1.2 Monotonicity

As mentioned in Section 3.3.2, it is desirable to have monotonic clustering strategies as the reversals caused by non-monotonic strategies are inconvenient and hard to interpret [Murt 85, Morg 95]. We can give here a concrete example from the problem domain of cell detection: assume two keypoints merged in one cluster with distance 0.7. Since this distance is more than 0.5, they will belong to different cells. In a non-monotonic linkage, you may encounter a case where you merge a third keypoint to the aforementioned cluster with distance less than 0.7, say 0.4. Consequently, the cluster formed by the three keypoints will form a single cell. This result is not consistent with the fact that the first two keypoints were dissimilar and it is thus hard to know which of the two decisions is correct.

As also mentioned in Section 3.3.2, when β_4 is zero, the linear model Π given by Eq. 5.10 is monotonic under the condition $\beta_1 + \beta_2 + \beta_3 \geq 1$. Obviously, this is fulfilled by our customized linkage and the trees built by $\Pi_{\text{Customized}}$ are thus monotonic.

5.5.2 Finding the Hit-point

Applying the hierarchical clustering will result in different clusters of keypoints. Each cluster $\mathbb{A}$ represents a cell. In order to determine a single hit-point, we use the following equation:

$$\mathbf{p}_h = \frac{1}{\sum_{j=1}^{|\mathbb{A}|} \text{rw}_j} \sum_{l=1}^{|\mathbb{A}|} \text{rw}_l \mathbf{p}_l, \tag{5.17}$$

where $|\mathbb{A}|$ is the number of keypoints inside the considered cluster, $\mathbf{p}_l$ is the keypoint l in the cluster, and rw_l is a reliability weight that we set to $|\mathrm{DoG}_{\gamma 1}(\mathbf{p}_l)|$.

5.6 System Training

Equipped with the previously described methods, one is now able to train the proposed system from a given set of images and their cell segmentation (ground truth). First, the system is calibrated to detect bright or dark cells according to Eq. 5.1. Depending on the result of this step, the keypoints with positive or negative $\mathrm{DoG}_{\gamma 1}$ values will be discarded. After that, a SIFT threshold is learned from the training data and then applied as described in Section 5.3. Using the remaining set of keypoints, the mean keypoint scale $\bar{\sigma}$ is computed and SAS is applied as described in Section 5.4.3. Next, keypoints are detected in all training images and their respective features are extracted. Based on the ground truth, the class of each keypoint can be determined. Using this information a balanced random forest is trained on the keypoint features. These steps enable the system to automatically extract stable keypoints and to determine whether they belong to a cell or the background.

Based on the cell keypoints, the profile learning process can then be started. As a first step, we compute the maximum inner profile length MaxL from the training images and their ground-truth masks as described in Section 5.4.2. Next, we train a balanced random forest using the profile features. As classes, we use *inner profile* and *cross profile*. This yields a system that is automatically able to distinguish these two types of profiles.

For the hierarchical clustering, no training is required. The method can be applied directly on the probabilistic output of the profile random forest.

5.7 Evaluation Measures

We use the following measures to evaluate detection quality: precision, recall, detection error, F-measure, detection time, and centeredness error. Recall addresses the following question: from 100 cells, how many of them are detected? On the other hand, precision answers the question: from 100 hit-points (cf. Eq. 5.17), how many of them are correct? A question which arises in this regard is how to deal with over-detected cells, i. e. cells which are detected by several hit-points? Does an over-detected cell count when calculating the recall, i. e. do we consider it *detected*? On the other hand, assuming an answer "no" to the previous question, should these over-detection points (inside an over-detected cell) reduce precision as well? Typically, the over-detection points may not be counted twice. In other words, they should degrade either precision or recall but not both. The concept of precision, however, is more appropriate to quantify over-detection, and it is thus usually used for this purpose. In summary, an over-detected cell will be considered *detected* when computing the recall, but its over-detection points will reduce precision.

Detection error of a cell detection algorithm applied on image I is defined in this thesis as the arithmetic average of the precision loss and the recall loss:

$$\text{Detection error}(I) = \frac{1}{2}\left(\frac{N_C^U}{N_C} + \frac{N_H^B + N_H^O}{N_H}\right), \tag{5.18}$$

where N_H is the total number of hit-points. N_H^B is the number of hit-points in the background. N_H^O is the number of over-detection hit-points. For instance, if five hit-points were detected in one cell, then one of them is considered correct and the other four are considered over-detection hit-points. N_C is the total number of cells in the considered image I. N_C^U is the number of undetected cells, i. e. the cells which contain no hit-points.

On the other hand, F-measure is obtained by computing the harmonic mean of precision and recall instead of computing the arithmetic mean of their losses. If one of the values of precision or recall is low, then the harmonic mean is closer to the minimum than the arithmetic average. In general, however, when the values of precision and recall are available, it does not matter much which of the two averages to use. Later on in this chapter and in the next two chapters, we compare our algorithms with other methods in literature. We choose the appropriate measure so that it conforms to the available published results by the other methods.

Since a hit-point can lie anywhere inside a cell mask, a measure is needed in order to evaluate the centeredness of the hit-point inside a detected cell. For this purpose, we define a new measure:

$$\text{Centeredness error}(I) = \frac{1}{N_C^C}\sum_{j=1}^{N_C^C}\frac{\|\mathbf{p}_h^j - \mathbf{p}_m^j\|}{\kappa_j}, \tag{5.19}$$

where N_C^C is the number of correctly detected cells, i. e. the cells which were detected by only one hit point. Therefore, over-detected cells are not considered by this measure. The numerator is the Euclidean distance between $\mathbf{p}_h^j$ the hit-point inside the cell j and $\mathbf{p}_m^j$ the center of mass of this cell. The denominator κ_j is the major axis length of the ellipse which represents the covariance matrix of the binary mask of the cell j. This normalization is important in order to make the centeredness error independent of cell size.

5.8 Evaluation

In Section 5.8.1, all cell lines of Table 4.1 were used to evaluate the detection accuracy. The real cell lines were used in Section 5.8.2 to assess the contribution of the different components of our algorithm to the overall detection accuracy. In Section 5.8.3, we perturb the three real cell lines with orientation, scale, and illumination changes in order to assess the system's invariance to these factors. Detection time was evaluated in Section 5.8.4. In Section 5.8.5, we compare our system with two other approaches on bright field microscopy. For this comparison, the CHO cell line was used first as is, and then perturbed with illumination and scale changes. In Section 5.8.6, we show qualitative evaluation on images produced by COSIR hardware. The ability of the

algorithm to learn to detect cells when the training data contains several cell lines was evaluated in Section 5.8.7. Lastly, we apply the proposed algorithm on phase contrast datasets in Section 5.8.8.

	Precision	Recall	Detection error	Centeredness error
CHO	77.7 ± 8.0	92.9 ± 3.0	0.147 ± 0.03	0.477 ± 0.13
L929	82.8 ± 4.4	92.6 ± 2.9	0.123 ± 0.01	0.377 ± 0.07
Sf21	97.3 ± 0.9	96.4 ± 3.2	0.031 ± 0.01	0.164 ± 0.02
Simulated A	98.9 ± 0.7	99.4 ± 0.7	0.008 ± 0.01	0.173 ± 0.10
Simulated B	95.4 ± 2.1	96.5 ± 1.6	0.040 ± 0.01	0.222 ± 0.11

Table 5.1: Cross-validation estimates of cell detection accuracy on different cell lines: one image per cell line is used for training and the other images of the same cell line are used for testing. This was repeated for each image in the real cell lines and five times in the simulated cell lines.

5.8.1 Evaluation of the Overall Detection Accuracy

We evaluated detection accuracy of the proposed system on the five cell lines given in Table 4.1. For each cell line, only one image was used for training and the others were used for testing. For the real cell lines, this was repeated for each image. For the simulated cell lines, this was repeated five times. Table 5.1 shows the results of this cross validation.

Table 5.1 shows that the error was close to zero for the high SNR simulated images. Even under severe Gaussian noise conditions with SNR ≈ 0.07, the error was only 4 %. It is also clear from the table, that the system achieved higher detection rates with suspension cells compared to adherent cells. This is plausible, as the latter have considerably lower contrast than suspension cells.

5.8.2 Evaluation of the System Components

We also evaluated the contribution of specific components of the system to the detection accuracy. Tables 5.2, 5.3, and 5.4 summarize this evaluation on the L929, CHO, and Sf21, respectively. In the first row, the outputs of the two random forests were combined using the connected components as described in Section 5.5. The same was done in the second row but the keypoints were thresholded according to Section 5.3. This thresholding is also used in rows 3 to 8. In the third row, the agglomerative hierarchical clustering with the group average linkage was used instead of the connected components. In the fourth one, our customized linkage method was used instead. This customized linkage is also used in rows 5 to 8. In the fifth row, Eq. 5.17 was used to find the hit-points instead of the simple arithmetic average of the coordinates of the keypoints inside each cluster. Eq. 5.17 is also used in rows 6 to 8. In the profile expansion rows, three strategies were tested: 1) using parallel profile sets, 2) SAS, 3) using profile sets together with SAS.

The estimates in tables 5.2, 5.3, and 5.4 are cross validation estimates, where one image per cell line is used for training and the remaining ones are used for testing.

		Precision	Recall	Detection error	Centeredness error
CC		67.1 ± 4.2	66.2 ± 5.9	0.334 ± 0.04	0.528 ± 0.08
SIFT threshold		76.8 ± 4.2	73.2 ± 3.4	0.250 ± 0.01	0.381 ± 0.09
AHC average		72.6 ± 5.1	91.0 ± 4.5	0.182 ± 0.01	0.484 ± 0.11
AHC custom		76.0 ± 4.5	90.2 ± 4.4	0.169 ± 0.01	0.406 ± 0.08
Weighted avg.		76.9 ± 4.4	91.0 ± 4.6	0.161 ± 0.01	0.382 ± 0.07
Profile expansion	Profile sets	78.9 ± 5.1	91.7 ± 4.0	0.147 ± 0.02	0.382 ± 0.07
	SAS	82.8 ± 4.4	92.6 ± 2.9	0.123 ± 0.01	0.377 ± 0.07
	Both	83.4 ± 4.0	92.6 ± 3.0	0.120 ± 0.01	0.369 ± 0.07

Table 5.2: Contribution of the different system components to detection accuracy (L929). The estimated measures are obtained as follows: one image is used for training and the other images are used for testing. This is then repeated for each image in a cross-validation loop.

		Precision	Recall	Detection error	Centeredness error
CC		70.5 ± 5.3	58.7 ± 6.4	0.354 ± 0.05	0.472 ± 0.20
SIFT threshold		76.8 ± 5.9	65.5 ± 4.7	0.288 ± 0.04	0.432 ± 0.07
AHC average		68.5 ± 10.7	91.4 ± 3.9	0.201 ± 0.03	0.652 ± 0.24
AHC custom		73.4 ± 8.8	90.1 ± 4.4	0.183 ± 0.02	0.503 ± 0.14
Weighted avg.		74.0 ± 8.6	90.7 ± 4.8	0.177 ± 0.02	0.501 ± 0.14
Profile expansion	Profile sets	76.4 ± 8.4	91.9 ± 4.3	0.159 ± 0.02	0.455 ± 0.10
	SAS	77.7 ± 8.0	92.9 ± 3.0	0.147 ± 0.03	0.477 ± 0.13
	Both	78.8 ± 7.8	93.1 ± 3.2	0.141 ± 0.03	0.446 ± 0.11

Table 5.3: Contribution of the different system components to detection accuracy (CHO). The estimated measures are obtained as follows: one image is used for training and the other images are used for testing. This is then repeated for each image in a cross-validation loop.

		Precision	Recall	Detection error	Centeredness error
CC		89.0 ± 2.2	78.6 ± 2.4	0.162 ± 0.02	0.132 ± 0.00
SIFT threshold		93.1 ± 1.9	83.3 ± 4.1	0.118 ± 0.03	0.180 ± 0.01
AHC average		90.9 ± 1.6	95.8 ± 0.9	0.066 ± 0.01	0.216 ± 0.01
AHC custom		94.1 ± 0.9	95.1 ± 0.9	0.054 ± 0.00	0.189 ± 0.02
Weighted avg.		94.3 ± 0.8	95.2 ± 1.2	0.053 ± 0.01	0.163 ± 0.01
Profile expansion	Profile sets	95.0 ± 1.5	96.3 ± 1.1	0.043 ± 0.01	0.157 ± 0.01
	SAS	97.3 ± 0.9	96.4 ± 3.2	0.031 ± 0.01	0.164 ± 0.02
	Both	97.4 ± 0.9	96.8 ± 3.1	0.029 ± 0.01	0.162 ± 0.02

Table 5.4: Contribution of the different system components to detection accuracy (Sf21). The estimated measures are obtained as follows: one image is used for training and the other images are used for testing. This is then repeated for each image in a cross-validation loop.

Cell line	Keypoint features	Profile features
CHO	$\mathrm{Ray}_n(\mathbf{p}, 180°)$	$\mathrm{mean}(\frac{d^2\mathrm{prof}}{d\chi^2})$
L929	SIFT descriptor feature	V_2
Sf21	SIFT descriptor feature	V_2
Simulated A	Stencil feature	$\mathrm{mean}(\frac{d^2\mathrm{prof}}{d\chi^2})$
Simulated B	$\mathrm{VMap}(\mathbf{p}, 6)$	V_2
All	$\mathrm{DoG}_{\gamma 1}$	$\mathrm{mean}(\frac{d^2\mathrm{prof}}{d\chi^2})$

Table 5.5: Highest ranked features according to the following measure: OOB error increase induced by random permutation of feature values. For each cell line, a random forest was trained using a randomly chosen image from the considered cell line. In the last row, the five randomly chosen images (image from each cell line) were used for training.

An important result of these tables is that SAS achieves higher detection scores than the profile set. On the other hand, using SAS together with the profile set delivers a bit higher detection rate than using SAS alone. Nevertheless, we sacrifice this small improvement in the detection rate in favor of better detection times. All other experiments in this thesis were done with SAS alone. Therefore, the SAS rows in tables 5.2, 5.3, and 5.4 correspond to the estimates in Table 5.1.

As shown in Figure 4.1, simulated cells tend to form negative $\mathrm{DoG}_{\gamma 1}$ blobs, whereas real cells tend to form positive $\mathrm{DoG}_{\gamma 1}$ blobs in the positively defocused images. The system was able to automatically detect the right type of blobs in each case using Eq. 5.1.

We did not conduct a thorough analysis of feature importance. However, the random forest has an internal mechanism to rank its features: the OOB error increase due to random permutation of feature values (cf. Section 3.2.2). A random image from each cell line was used to train the system and the feature with the highest rank was recorded. Table 5.5 shows the results. The last row displays the highest ranked feature when the previous five images (image from each cell line) are used together to train the system. According to this table, at least one keypoint feature from each of the five keypoint feature sets was ranked the best. This does not imply that all feature sets are necessary to obtain the detection rate reported in Table 5.1. However, it gives an indicator that all keypoint feature sets are informative. Regarding profile features, Table 5.5 shows that the mean value of the second derivative and V_2 (cf. Eq. 5.5) are the two highest ranked features.

5.8.3 Evaluation of Illumination, Orientation, and Scale Invariance

The simulation software in [Lehm 07] can generate illumination artifacts on simulated images. In order to make the experiment more realistic, we applied the simulated illumination field on our real cell lines. Figure 5.7 shows how the detection error changes with the illumination scale, i. e. the ratio between illumination energy and image energy. Figure 5.8 shows the difference between an image at illumination scale zero and another one at illumination scale 100. The detection error change between these two extreme cases, as shown in Figure 5.7, was 8% in the worst case.

Only one image (randomly chosen) in each cell line was used for training. It is an image at illumination scale zero. The other images in the cell line were used for testing at each of the following illumination scales: 0, 20, 40, 60, 80, and 100.

Figure 5.9 depicts the system's invariance to image scale. In this experiment, the system was trained using one randomly chosen image per cell line and tested on other up- or down-sampled images from the same cell line. The figure shows that the detection error change is in the range of 4% excluding a sudden increase in the detection error of Sf21 at scale 0.5. This reduction in detection error at scale 0.5 is, in fact, a reduction in recall only (not shown in the figure). This indicates that scale invariance is limited from the bottom. The reason is that downsampling reduces the number of SIFT keypoints due to structure degradation which causes a deterioration in detection recall.

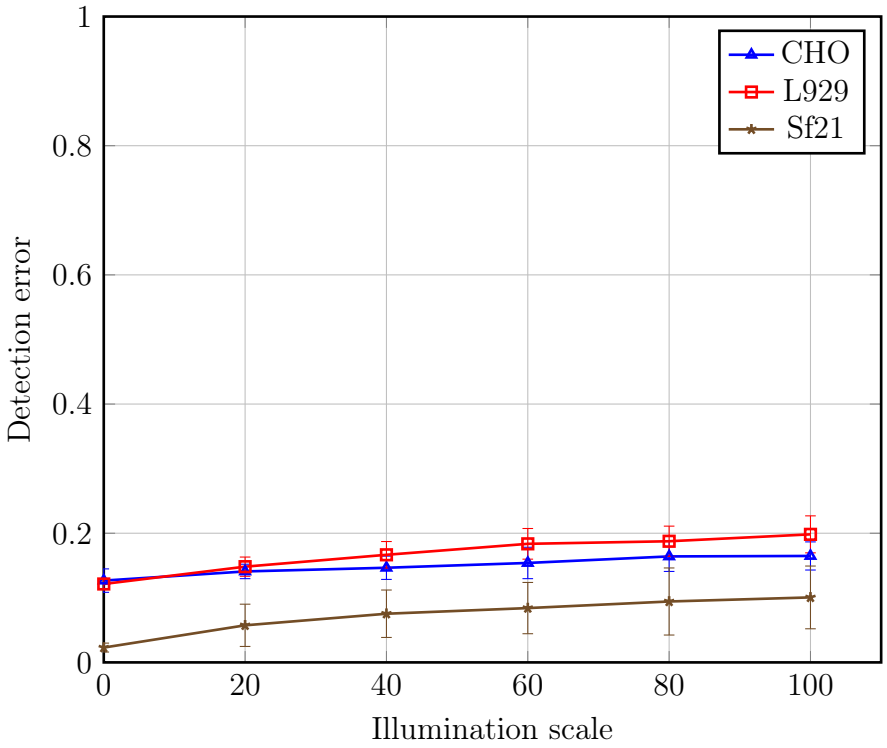
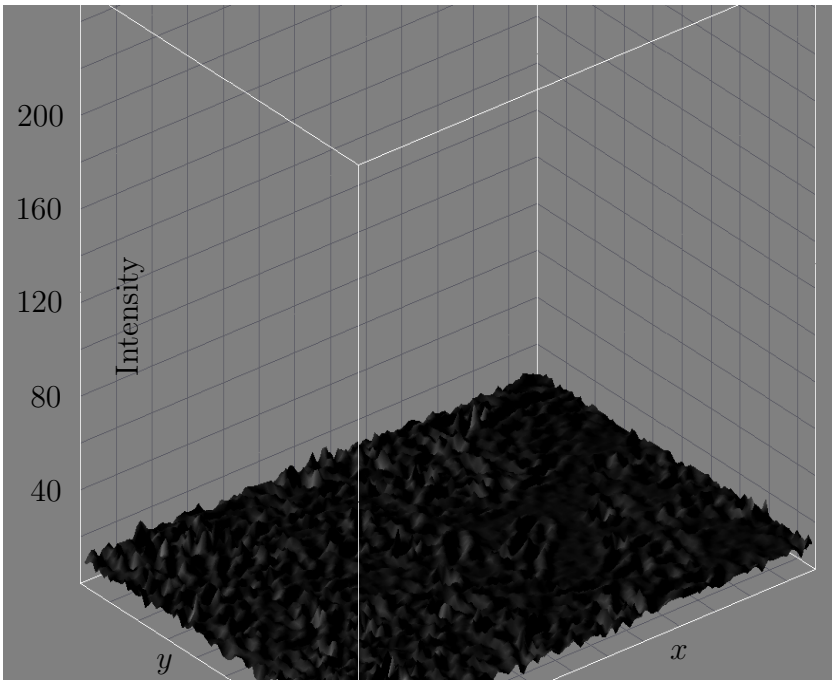


Figure 5.7: Illumination invariance: for each cell line, an image (randomly chosen) at illumination scale 0 was used for training and the other images of the same cell line were used for testing at different illumination scales.

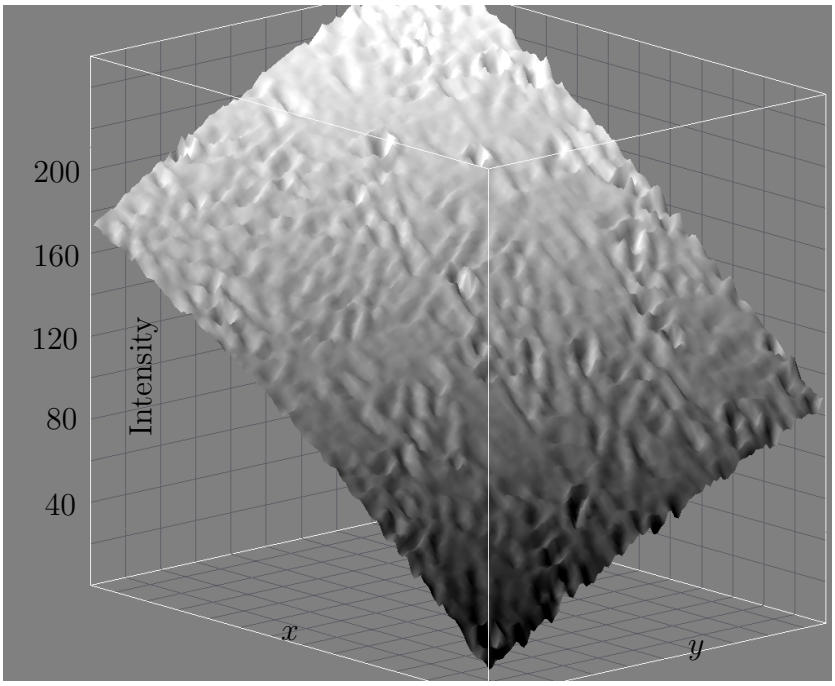
Finally, Figure 5.10 shows the degree of invariance with respect to cell orientation. Rotating the images in order to test the system invariance to the orientation change is not the best choice because of the diversity of cell orientations in each image. Therefore, we simulated cell images so that all cells inside the same image have the same orientation. The shape model in the simulation software [Lehm 07] cannot generate elongated cells with dominant orientation. Therefore, for this experiment, we replaced its shape model with an elliptical one. The system was trained using one randomly chosen image at orientation zero and tested on other five images at each of the following orientations: 0°, 30°, 60°, 90°, 120°, and 150°. Each image contained 150 simulated cells.

5.8.4 Evaluation of the Detection Time

In order to investigate the feasibility of the algorithm, we measured the detection time for all cell lines. The evaluation was done on a Dell laptop with 8 GB RAM and an Intel Core i7-2720QM processor with clock speed 2.20 GHz. The implementation details are as follows: the feature extraction was implemented in Matlab, the classification was done using the R package randomForest [Liaw 02], the agglomerative hierarchical clustering step was implemented in Java, and SIFT features were obtained from VlFeat [Veda 08] (C with Matlab interface). All modules were put together in a single Matlab application.



(a) CHO image at illumination scale = 0



(b) CHO image at illumination scale = 100

Figure 5.8: Illumination invariance example: the upper figure exemplifies a training image in the illumination invariance experiment, whereas the lower one is an example of a testing image. In both figures, the intensity is plotted as a function of the spatial dimensions x and y.

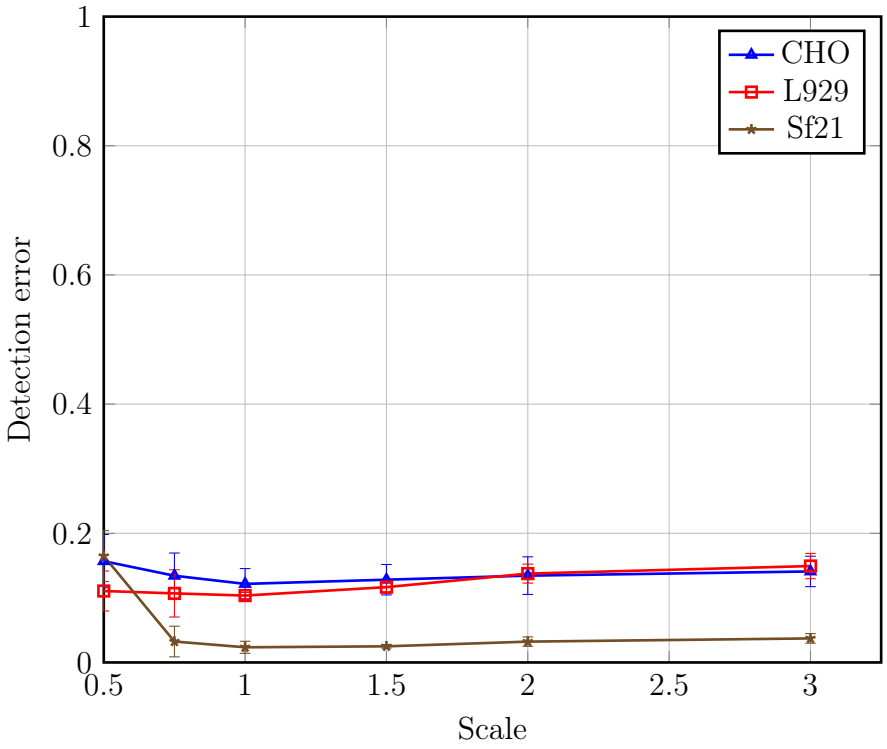


Figure 5.9: Scale invariance: for each cell line, an image (randomly chosen) at scale 1 was used for training and the other images of the same cell line were used for testing at different scales.

	Original images		Subsampled images	
	Detection time	Detection error	Detection time	Detection error
CHO	45.88 ± 13.60	0.147 ± 0.03	13.75 ± 2.11	0.128 ± 0.01
L929	36.69 ± 5.59	0.123 ± 0.01	12.68 ± 0.93	0.112 ± 0.01
Sf21	40.65 ± 7.13	0.031 ± 0.01	8.97 ± 1.07	0.071 ± 0.02
Simulated A	30.47 ± 0.33	0.008 ± 0.01	7.91 ± 0.34	0.009 ± 0.00
Simulated B	31.00 ± 1.88	0.040 ± 0.01	4.93 ± 0.26	0.043 ± 0.00

Table 5.6: Evaluation of the detection time (in seconds per image): resolution of CHO, L929, and Sf21 images is 1280 × 960 pixels. Resolution of Simulated A and Simulated B is 1200 × 1200 pixels. All resolutions are given before subsampling. Estimates are generated using a cross-validation loop compatible with Table 5.1.

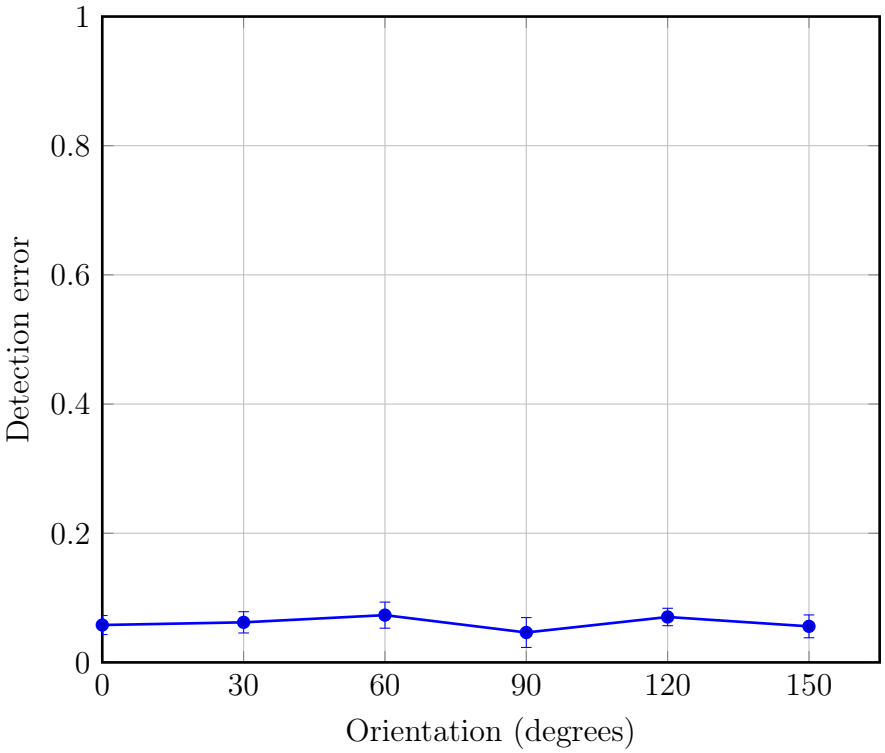


Figure 5.10: Orientation invariance: an image (randomly chosen) at orientation 0 was used for training and other five images at each orientation were used for testing. Each image contains 150 cells simulated under an elliptical shape model.

The system was trained and tested as described in Section 5.8.1. The detection times are reported at the left-hand side of Table 5.6. The right-hand side of the table shows the results when the same experiment was applied on subsampled images (subsampling factor 0.5). As can be seen in the table, the detection time is approximately in the range $[30, 46]$ seconds per image and drops to the range $[5, 14]$ seconds per image after subsampling.

5.8.5 Comparison with Other Approaches in Bright Field Microscopy

We compared our system with [Beca 11] and [Ali 12] which were developed specifically for bright field microscopy (cf. Section 1.2). Table 5.7 summarizes the required input for each approach. In [Ali 12], there is a well-developed segmentation approach. It is, however, worth pointing out that only the detection part is used in the comparison. [Beca 11] utilizes three algorithms at three different focus levels and combines the results. In our evaluation, instead of combing the results of these three algorithms, we select the one which has the minimum error. This strategy gave better results on our images.

Due to the difficulty of the manual parameter tuning, this comparative evaluation was performed using only one cell line, CHO, and without cross validation. One image was randomly chosen and used to train our system. The same image was used

for parameter tuning of [Beca 11] and [Ali 12]. The rest of the images were used for testing.

The software of [Beca 11] was obtained from its authors while we implemented the cell detection part of [Ali 12] ourselves. The optimal value of the single parameter of [Ali 12] was found by scanning the parameter domain and selecting the value which minimizes the detection error. On the other hand, we optimized the parameters of [Beca 11] manually. The result of the comparison is shown in the upper part of Table 5.8.

In order to investigate how the three approaches perform under illumination and scale change, we did the following experiment: the comparison was applied on the CHO cell line, but after perturbing the images. An illumination field was applied on all CHO images. The same field was applied on all testing and training images. In addition, the testing images were resampled using the following scales: 0.5, 0.75, 1, 1.25, and 1.5. Training of the proposed approach and parameter fine-tuning of [Beca 11] and [Ali 12] were performed again on the perturbed training image. The results are shown in the lower part of Table 5.8.

	[Ali 12]	[Beca 11]	Our approach
Required number of images	2	3	1
Manually tuned parameters	1	> 9	0

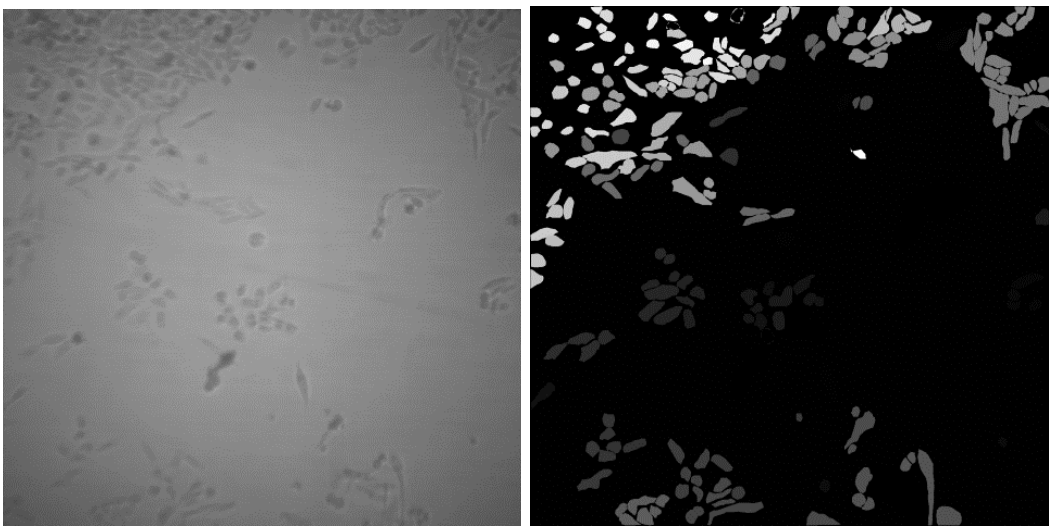
Table 5.7: Input requirements for [Ali 12], [Beca 11], and the proposed supervised approach

		Precision	Recall	Detection error	Centeredness error
CHO	Proposed method	88.1 ± 2.3	87.6 ± 4.2	0.122 ± 0.02	0.373 ± 0.34
	[Ali 12]	56.1 ± 11.1	91.8 ± 3.5	0.260 ± 0.07	0.552 ± 0.40
	[Beca 11]	80.9 ± 3.2	61.3 ± 9.3	0.288 ± 0.04	0.495 ± 0.58
CHO perturbed	Proposed method	80.8 ± 3.7	91.4 ± 2.5	0.138 ± 0.02	0.469 ± 0.28
	[Ali 12]	81.0 ± 12.7	36.5 ± 18.9	0.412 ± 0.06	0.665 ± 0.73
	[Beca 11]	43.0 ± 16.2	23.4 ± 10.6	0.668 ± 0.04	1.350 ± 2.03

Table 5.8: Comparison with other approaches specifically developed for bright field microscopy: in the upper part of the table, all approaches were applied to the CHO images. In the lower part of the table, the same experiment was repeated after perturbing the images by illumination and scale changes.

5.8.6 Qualitative Evaluation on COSIR Images

As mentioned in the introduction of this thesis (cf. Section 1.1), our research was conducted in context of the interdisciplinary research project COSIR. In this section, we show a qualitative evaluation of our supervised cell detection pipeline on COSIR images. These images were described in Section 4.2. The algorithm was trained using one image (cf. Figure 5.11) acquired from an arbitrarily-chosen channel of the 24 channels of a COSIR system. The trained algorithm was then applied on images obtained from other channels. Figure 5.12 exemplifies detection results.



(a) The COSIR image which was used for training

(b) Ground truth of the image at the left-hand side

Figure 5.11: Evaluation on COSIR images: a COSIR image and its ground-truth mask were used to train the proposed supervised cell detection pipeline.

5.8.7 Evaluation of the Generalization on Multiple Cell Lines

In Section 5.8.1, the system was trained separately for each cell line. In fact, it is more challenging to learn to detect cells when images from different cell lines are used in the training. In this experiment, one image from each cell line was randomly chosen. The five chosen images were used to train the system. The rest of the images were used for testing. This process was repeated five times. As the images are of different dynamic ranges and/or modalities, each image was normalized to $[0, 1]$. The simulated images were also inverted in order to have one-sided cell keypoints in the training data. Table 5.9 shows the results. Comparing Table 5.9 to Table 5.1, one can see a relatively considerable increase of the detection error for Sf21 and Simulated B. Nevertheless, the maximum detection error is still 15.5%.

In order to investigate whether similar cells are more suited for joint training, we conducted two additional experiments. Table 5.10 shows the results of the same process described above, but training and testing were applied only on the adherent cell lines. The same applies for Table 5.11, but for the simulated cell lines. The detection error in both tables is very close to the detection error in Table 5.1.

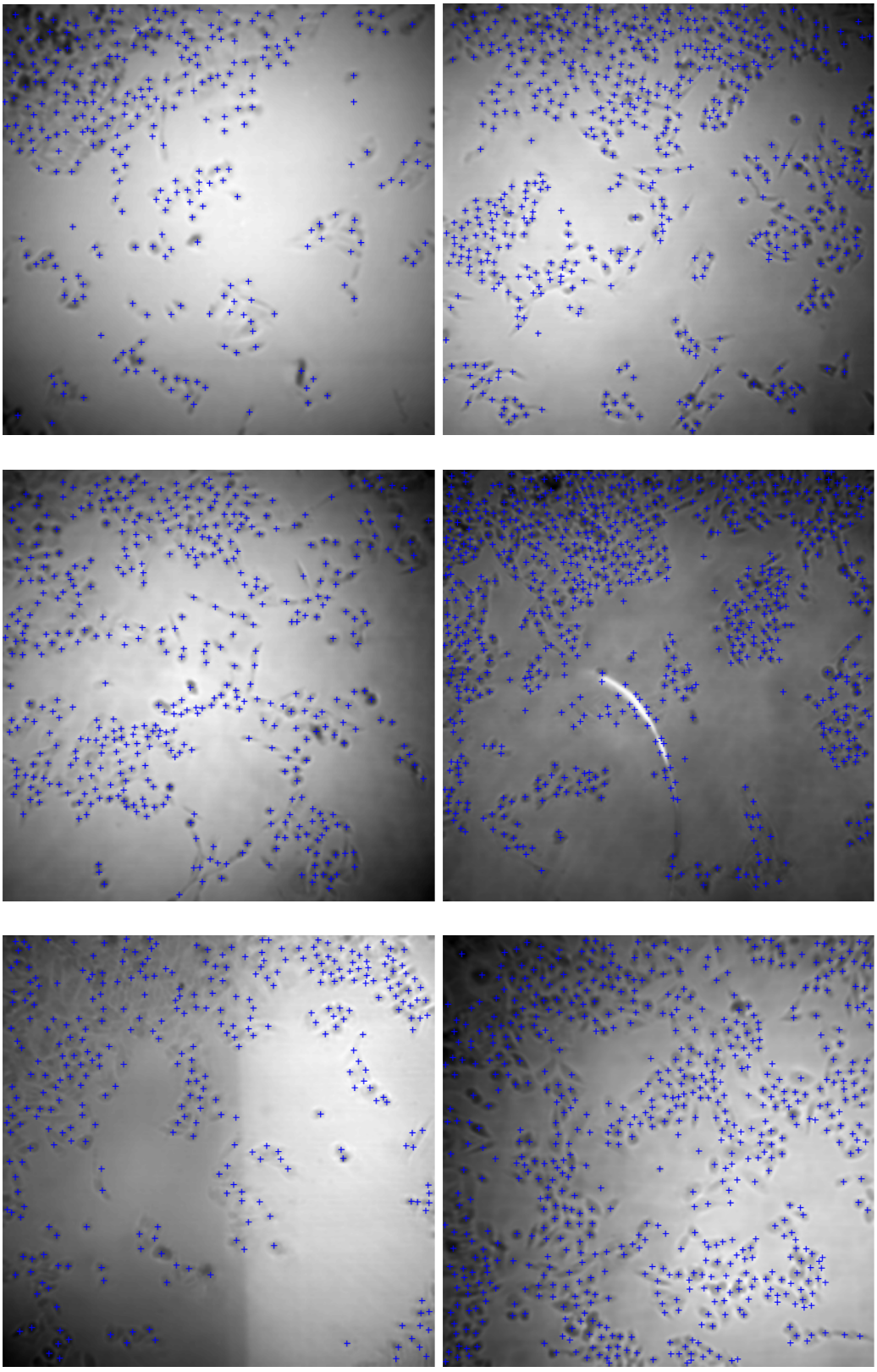


Figure 5.12: Evaluation on COSIR images: the algorithm, after training on one image (cf. Figure 5.11) obtained from an arbitrarily-chosen channel, was applied on images acquired from other channels.

	Precision	Recall	Detection error	Centeredness error
CHO	77.7 ± 7.2	92.2 ± 4.3	0.150 ± 0.02	0.498 ± 0.08
L929	79.6 ± 8.0	89.4 ± 6.6	0.155 ± 0.01	0.438 ± 0.12
Sf21	73.2 ± 2.4	99.7 ± 0.1	0.136 ± 0.01	0.179 ± 0.04
Simulated A	98.9 ± 0.1	98.3 ± 0.4	0.014 ± 0.00	0.177 ± 0.00
Simulated B	81.8 ± 5.7	97.6 ± 0.3	0.103 ± 0.03	0.256 ± 0.01

Table 5.9: Joint training: five images were randomly chosen, one from each cell line. They were used to train the system and the rest were used for testing. This process was repeated five times.

	Precision	Recall	Detection error	Centeredness error
CHO	76.8 ± 5.0	94.3 ± 2.2	0.145 ± 0.01	0.497 ± 0.13
L929	80.7 ± 5.1	92.0 ± 3.6	0.137 ± 0.01	0.462 ± 0.06

Table 5.10: Joint training: two images were randomly chosen, one from CHO and another one from L929. They were used to train the system and the rest were used for testing. This process was repeated five times.

	Precision	Recall	Detection error	Centeredness error
Simulated A	98.8 ± 0.2	98.1 ± 0.4	0.015 ± 0.00	0.173 ± 0.00
Simulated B	93.3 ± 1.0	97.0 ± 0.3	0.048 ± 0.00	0.229 ± 0.01

Table 5.11: Joint training: two images were randomly chosen, one from Simulated A and another one from Simulated B. They were used to train the system and the rest were used for testing. This process was repeated five times.

5.8.8 Evaluation on Phase Contrast Datasets

	F-measure (%)	Time (seconds)	Centeredness error
Pan et al. [Pan 10]			
Trained on all TIs	94.4	900.0	-
Our approach			
Trained on 1st TI	89.4 ± 2.0	7.75 ± 0.98	0.137 ± 0.009
Trained on 2nd TI	89.6 ± 2.2	7.56 ± 1.17	0.136 ± 0.007
Trained on all TIs	90.4 ± 2.1	7.66 ± 1.18	0.132 ± 0.005
Our approach with upsampling			
Trained on 1st TI	90.6 ± 1.8	10.26 ± 1.24	0.136 ± 0.009
Trained on 2nd TI	92.8 ± 1.7	10.60 ± 1.27	0.138 ± 0.011
Trained on all TIs	92.1 ± 1.2	13.24 ± 1.84	0.148 ± 0.014

Table 5.12: Comparison of the proposed supervised pipeline with [Pan 10] on the Bovine phase contrast dataset. TI is an abbreviation of *Training Image*. In this experiment, training images belong to a single temporal image sequence.

Table 4.2 contains a summary of the available phase contrast datasets. It is not possible to train our supervised approach on the dataset given by [Arte 12] because its ground truth contains only cell centers (no border delineation). Therefore, only the dataset given in [Pan 10] (cf. Table 4.2) was used in this section. The first row of Table 5.12 shows detection accuracy of the cell detection approach published in [Pan 10]. These results were obtained by training the approach on 10 images and testing it on other 10 images (the 10 Bovine images in Table 4.2). We do not have access to the software of [Pan 10]. Therefore, in order to compare it with our approach, we used the same training and testing images used in [Pan 10]. The set of training images is an *image sequence*, i. e. it contains images of the same cell culture acquired at successive time steps. Instead of training our approach with the entire training dataset, we trained it on the first image of the training image sequence and discarded the rest of the training images. The resulting model was then applied on the test image set, i. e. on the 10 Bovine images of Table 4.2. The results are shown in the second row of Table 5.12. In the third row of Table 5.12, we see the results of the same procedure, but the first two images of the training image sequence are used for training. In the fourth row, the entire training dataset was employed for training.

We noticed that part of the detection error in our approach is due to the fact that there are small cells, i. e. cells with very few pixels, in which SIFT was unable to detect any keypoint. This conforms to the results of the scale-invariance experiment in Section 5.8.3, in which insufficient resolution degraded the detection recall. Therefore, we repeated the evaluation of our approach on the phase contrast dataset, but on upsampled versions (upsampling factor is 2) of the training and testing images. The results are shown in the lower part of Table 5.12, i. e. in the fifth, sixth, and seventh row. We point out that in some SIFT implementations, this upsampling is provided as an option by which it is possible to detect fine structures which are undetectable when the Gaussian pyramid is started from the original resolution. See, for instance, the *first octave* option in VlFeat [Veda 08].

The results show that our F-measure is, although improved with upsampling, but still slightly inferior to the results obtained by [Pan 10]. Our approach is, however, able to achieve a decent F-measure value even when trained with a single image. Moreover, the proposed approach seems to be much more efficient in terms of detection time as it is about two orders of magnitude faster than [Pan 10].

5.9 Discussion

As mentioned in Section 1.2, several approaches [Ali 12, Ali 07, Beca 11] utilize, though in different ways, images at two or three focus levels to improve contrast. This is appealing as it is possible to get considerably higher contrast from the intensity change with the defocus distance. The drawback is that at least two images are needed. As this is not always available, the algorithm presented in this chapter was developed to work with a single defocused image. In fact, using one image has another important advantage: it facilitates extending the algorithm in the future for more image modalities which are, in general, not expected to expose the same behavior of bright field microscopy with defocussing. Nevertheless, in Chapter 6, we will see how we can go further by utilizing multiple images.

The approach in [Pan 10] on phase-contrast microscopy has a partial conceptual similarity with the algorithm presented in this chapter. This similarity lies in the use of two classifiers which have goals analogous to the goals of our keypoint and profile classifiers. However, our features were carefully designed and heavily tested for the rotation-, scale-, and illumination-invariance. Another difference is that the whole system is fully automatic and its internal details, e. g. learning the maximum inner profile length, were designed for automation and invariance. In addition, the proposed approach is more efficient with respect to runtime as it is about 2 orders of magnitude faster. Moreover, the use of hierarchical clustering in order to optimally aggregate the second classifier results was a novel contribution which proved to be effective, especially using our customized linkage method. Lastly, the system is adapted to low contrast bright field microscopy. In fact, compared to the bright field approaches in [Beca 11] and [Ali 12] which need both multiple images and manual parameter tuning, our approach delivered higher detection accuracy and more robustness with respect to illumination and scale changes.

Before using the classifiers, the system must be trained. For the training, a set of images with ground truth is required. Due to the invariance of the extracted features and the use of random forests, the system can learn from a relatively small amount of training data. Only one image per cell line was used for training in our experiments.

Cell keypoints have either positive or negative $\mathrm{DoG}_{\gamma 1}$ values when they form valley-like or mountain-like structures, respectively. One can also notice that keypoints at cell boundaries and keypoints in cell interiors tend to have opposite $\mathrm{DoG}_{\gamma 1}$ signs (cf. Figure 5.2a). For the cell detection problem, however, cell interior keypoints are of main concern. We found in preliminary experiments (data not shown) that using one-sided keypoints leads to a better profile learning. For this reason, we consider either the positive $\mathrm{DoG}_{\gamma 1}$ keypoints or the negative, but not both.

In [Jurr 10], the intensity is sampled using a stencil instead of a patch and used for neuron detection in electron microscopy. A similar idea was used in a completely

different field [Mitt 10], where a radial sampling pattern was employed to sample 3D vessels. Both methods used fixed-size stencils. In our case, the scale and orientation of the keypoint deliver additional information. This can then be used to make the stencil scale- and orientation-invariant. In order to make it invariant to the local shift of intensity, one can subtract the minimum or the mean intensity of the stencil from all stencil nodes. We chose the mean because it is less sensitive to outliers.

In [Smit 09], it was shown that ray features are better than Haar-like features for cell recognition. Thus, we used ray features in our work. Our contribution to this feature set is that we made it scale- and orientation-invariant. In order to achieve scale-invariance for such features, computing the distances with respect to scale is insufficient. In fact, the gradient computation has to be additionally performed with respect to scale. If the gradient is computed using conventional kernels like Sobel or Prewitt, its norm Ray_n will be scale-dependent. We use a gradient computation that handles this issue correctly.

In [Wu 95], it was shown that variance maps can distinguish cells from background. The variance map value at a pixel is simply the variance of intensities in a neighborhood centered at this pixel. The neighborhood size is fixed and based on the cell size [Wu 95]. We extended this concept and made it scale-invariant.

If cells are circular and noise-free, we may get almost one keypoint per cell. Therefore, in such case, we will get very few or even zero inner profiles for training. If the cells are too far from each other, we may get very few or even zero cross profiles for training. This occurs because the algorithm extracts profiles between nearby keypoints only. Both cases occurred in some preliminary experiments (data not shown). The algorithm handles these two cases which makes the training robust against odd situations in the training data.

A main disadvantage of using a set of profiles is the computational cost. Furthermore, it turned out that SAS leads to a higher detection rate as shown in the results section. This is, probably, due to the fact that SAS serves also as a preprocessing step. For instance, we noticed that the SAS improved the edges drastically. Consequently, the edge-based features such as ray features, gained higher discriminative power.

In our experiments, we achieved robustness to low frequency changes in illumination by using features that are locally invariant to an offset in intensity. In a small region of an image, this local shift of intensity can be interpreted as a constant. We regard this as an important feature of our system, as many real world images, including images acquired by COSIR hardware, suffer from inhomogeneous illumination. The algorithm's robustness with respect to illumination was shown in: 1) quantitative evaluation on standard bright field images with simulated illumination fields, 2) qualitative evaluation on COSIR images with real-world illumination artifacts. It is worth pointing out that images with illumination artifacts should be used without normalization, because the image measures which are usually used in normalization like mean, standard deviation, maximum, and minimum of the image intensity depend on the illumination information.

If the distance between the two keypoints of a profile is large, then the illumination artifacts at this profile cannot be approximated by a constant. In other words, the invariance to local-intensity shift is beneficial only if the profiles are extracted between

nearby keypoints. Therefore, the proposed approach should somehow avoid extracting profiles between far keypoints. In order to achieve this goal, the algorithm learns the maximum inner profile length MaxL in a scale-independent manner from training data. This contributes to the partial illumination invariance of the profile learning and reduces both detection and training times. In fact, learning MaxL is particularly useful because this length is a cell characteristic. Whereas, for instance, learning the cross profile length does not make sense because it depends on the distribution of cells in the cell culture.

We used random forest as a classifier model for both keypoint and profile learning. As mentioned earlier, the random forest does not need parameter tuning which is one of our design goals. It is also inherently a multi-class classifier. This makes extending the keypoint learning in further research for debris and agglomeration detection easier. The previous two points can be considered as advantages of the random forest over some other state-of-the-art classifiers such as the SVM. In fact, some empirical studies [Khal 11, Khos 07] showed that the random forest outperformed the SVM in terms of area under ROC (AUC) on imbalanced data, even though that was not always the case [Meye 03]. Moreover, the immunity of the random forest against overfitting grants the system the ability to learn from small training data sizes. This last point makes it favorable over classifiers such as probabilistic boosting trees [Tu 05].

There is, in general, no guarantee that the *cell* and *background* classes are balanced. The same applies for the *inner* and *cross* classes in the profile learning. The imbalance problem can be solved by using a balanced random forest or a weighted random forest [Chen 04]. They tend to produce similar ROC curves. We chose to use the balanced version as it is computationally more efficient and less vulnerable to noise [Chen 04].

The output of the two classifiers can be seen as a graph whose nodes are the cell keypoints and whose edges are the inner profiles. Therefore, the connected components of this graph can be regarded as the detected cells. However, our results show that higher detection rates can be achieved when the probabilistic output of the profile classifier is utilized as a similarity measure in an AHC step. Moreover, detection rate was further improved by using our customized linkage method where more application-specific information was involved. In fact, it is plausible to expect that the AHC is more robust than the CC because it starts aggregating the most reliable cases. Furthermore, the decision about the less reliable cases does not depend on a single classification but on the average of several profile classifications. In the customized linkage, keypoints information is incorporated in order to make this *averaged* decision even more robust.

We pointed out in the motivation to this chapter (cf. Section 5.1) that a good cell detection approach should facilitate tracking when the latter is required. The output of our system after the clustering stage is a set of keypoints where each one is assigned to a cell and equipped with a set of features. A subgroup of these features has been used in tracking applications. See, for example, the use of SIFT descriptor for tracking in [Jian 10].

We evaluated the system with respect to runtime on our standard bright field image database and on a subsampled version of it. We noticed that, on this data, the detection time of the subsampled images was 3-4 times shorter than the detection

time of the original images. Since subsampling reduces the number of pixels to 25 % of their original number, the obtained result suggests that detection time is proportional to the number of pixels in the considered image. Moreover, with the exception of Sf21, the detection error was not increased by subsampling. In fact, for the adherent cell lines, it was decreased. This is probably caused by the sampling-induced smoothing of the cellular details which could otherwise mislead the detection algorithm and degrade its precision. However, this observation cannot be generalized without taking the original image resolution into account. In our experiments, the real cell line images before subsampling had a resolution 0.49 μm/pixel.

The system was also tested for its generalization ability. The results show that it can learn to detect cells even when several cell lines of different visual appearance are used for training. However, the detection error was smaller when only similar cell lines are used. It is thus beneficial to train the algorithm independently on each cell line or on each group of similar cell lines. In the next chapter, we clarify the importance of generalization ability and introduce an elegant solution using phase-based features.

Chapter 6

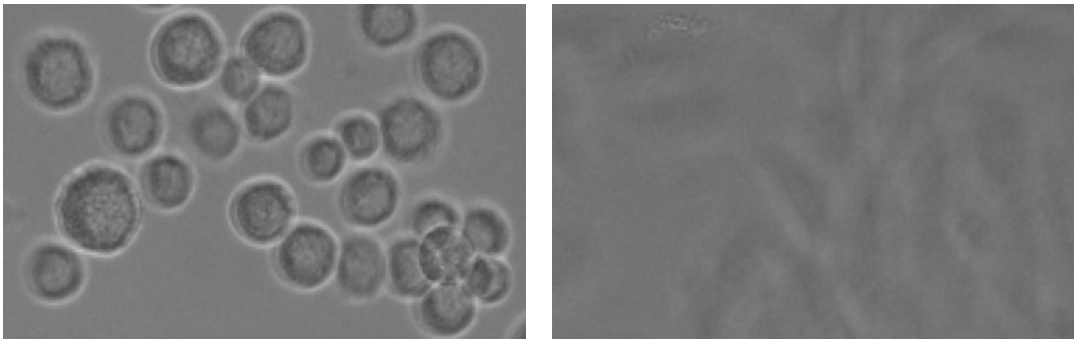
Improving Supervised Cell Detection Using Phase-based Features

Considerable parts of this chapter were already published in the following papers:
F. Mualla, S. Schöll, B. Sommerfeldt, S. Steidl, R. Buchholz, and J. Hornegger. "Improving joint learning of suspended and adherent cell detection using low-pass monogenic phase and transport of intensity equation". In: *Biomedical Imaging (ISBI), 2014 IEEE 11th International Symposium on*, pp. 927–930, Beijing, China, April 2014.
F. Mualla, S. Schöll, B. Sommerfeldt, A. Maier, S. Steidl, R. Buchholz, and J. Hornegger. "Using the low-pass monogenic signal framework for cell/background classification on multiple cell lines in bright-field microscope images". *International Journal of Computer Assisted Radiology and Surgery*, Vol. 9, No. 3, pp. 379–386, May 2014.
F. Mualla, S. Schöll, B. Sommerfeldt, and J. Hornegger. "Using the monogenic signal for cell-background classification in bright-field microscope images". In: *Proceedings des Workshops Bildverarbeitung für die Medizin 2013*, pp. 170–174, Heidelberg, Germany, March 2013.

6.1 Motivation

We mentioned in Section 1.2, that the low-pass monogenic signal was used as a boundary potential for cell segmentation in bright field microscopy. We also pointed out that the TIE was utilized for thresholding of bright field images and that the two concepts, i. e. the TIE and the low-pass monogenic signal, were linked in [Ali 10]. In this chapter, we explain this link in detail. We then show that local phase of the low-pass monogenic signal, which is not well-investigated in literature so far, improves the cell/background pixelwise classification. Afterwards, we employ both monogenic signal and TIE to improve joint learning, i. e. generalization ability (cf. Section 5.8.7), of our supervised approach presented in Chapter 5.

Section 5.8.7 conveys the message that the detection error of our supervised approach when trained using both suspended and adherent cell lines is higher compared to the case when the algorithm is trained separately for each cell line. This is probably due to differences in contrast and cellular details between adherent and suspended cells (cf. Figure 6.1). In this chapter, we show that it is possible to considerably improve the joint training of adherent and suspended cell lines, if a TIE-solution image



(a) Image of suspended Sf21 cells at defocus distance 15 μm

(b) Image of adherent CHO cells at defocus distance 30 μm

Figure 6.1: Illustration of differences in contrast and visual appearance between suspended and adherent cells. Defocus distances were experimentally chosen so that contrast is maximized. Original contrast was kept in this figure in order to show the difference in dynamic range.

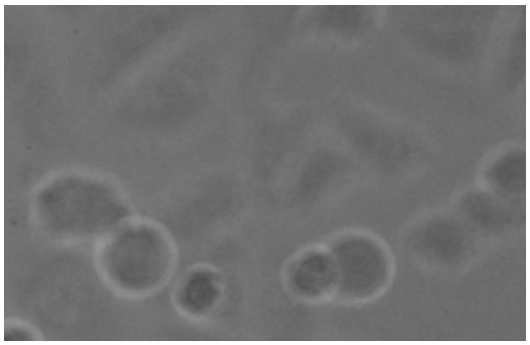


Figure 6.2: Illustration of coexistence of adherent and suspended cells in a cell culture

and/or local phase of the low-pass monogenic signal framework are used for feature extraction instead of a defocused image.

One may question the importance of this generalization ability as follows: it does not hurt to train the system on adherent cells alone and then applying the trained model on new images which contain only adherent cells. Similarly, this can be done for suspended cells. In fact, that what we did in all of our evaluations in Chapter 5 except Section 5.8.7. However, in general, suspended and adherent cells coexist in cell cultures (cf. Figure 6.2), even though this is not very well reflected in our image materials. Moreover, suspension and adherence can be seen as two terminal cases with shades of gray between them. Therefore, improving joint learning has a real practical importance.

The rest of this chapter is organized as follows: in Section 6.2, the closed-form expression of physical light phase as solution of the TIE after [Paga 98] is presented. In Section 6.3, the concept of local phase is explained at two levels: 1) *analytic signal* in one dimension, 2) *monogenic signal* in two dimensions. We show then in Section 6.4 how the monogenic signal was employed in [Ali 10] to approximate the TIE solution. In Section 6.5, we employ the TIE and monogenic signal to extend the supervised approach presented in Chapter 5 so that its joint learning capability

is improved. The evaluation is split in two main parts: Firstly, in Section 6.7.1, we compare the discriminative power of monogenic outputs with the discriminative power of defocused images. Secondly, in Section 6.7.2, we show that the pipeline suggested in Section 6.5 improves joint learning of suspended and adherent cells. The chapter is concluded by a discussion in Section 6.8.

6.2 Transport of Intensity Equation

As mentioned in Section 2.6, and repeated here for reader's convenience, TIE is a relation between the physical phase of light ϕ and the axial intensity derivative $\frac{\partial I}{\partial z}$ [Teag 83]:

$$-\frac{2\pi}{\lambda}\frac{\partial I(x,y)}{\partial z} = \nabla_\perp \cdot (I(x,y)\nabla_\perp \phi(x,y)), \tag{6.1}$$

where λ is the wavelength of light, z is the axial distance to the focus position, I is the intensity image at focus, $\nabla_\perp$ is the gradient operator in the two lateral dimensions x and y, i. e. inside the image plane, and $\nabla_\perp \cdot$ is the corresponding divergence operator.

The reader is referred to Section 2.6 for understanding the relation between ϕ and physical properties of the imaged objects. Based on [Paga 98], TIE can be analytically solved for ϕ as follows:

$$\phi = -\frac{2\pi}{\lambda}\nabla_\perp^{-2}\left(\nabla_\perp \cdot \left(\frac{1}{I}\nabla_\perp \nabla_\perp^{-2}\frac{\partial I}{\partial z}\right)\right), \tag{6.2}$$

where $\nabla_\perp^{-2}$ is the inverse Laplacian operator.

6.3 Monogenic Signal

6.3.1 One-dimensional case:

The monogenic signal is a 2D generalization of a fundamental concept in signal processing called analytic signal [Fels 01]. The latter is defined for a real-valued one-dimensional signal $g(x)$ by the following equation [Poul 10]:

$$g_{\mathfrak{a}}(x) := g(x) + i g_{\mathfrak{h}}(x), \tag{6.3}$$

where $i^2 = -1$ and $g_{\mathfrak{h}}(x)$ is the Hilbert transform of $g(x)$:

$$g_{\mathfrak{h}}(x) := \mathcal{H}(g(x)) := g(x) * \frac{1}{\pi x} = \frac{1}{\pi}\mathrm{CPV}\int_{-\infty}^{+\infty}\frac{g(\varrho)}{x-\varrho}d\varrho, \tag{6.4}$$

where CPV stands for the Cauchy principal value of the improper integral. In Fourier domain, it can be shown that Eq. 6.4 is equivalent to:

$$G_{\mathfrak{h}}(\omega) = -i\,\mathrm{sign}(\omega)G(\omega). \tag{6.5}$$

As can be seen in Eq. 6.5, the Hilbert transform of a signal is a phase shift of its frequency components by $\pm\frac{\pi}{2}$. Therefore, a signal and its Hilbert transform are commonly termed *quadrature pair*. Combining Eq. 6.3 and Eq. 6.5 yields:

$$\begin{aligned} G_{\mathfrak{a}}(\omega) &= G(\omega) + iG_{\mathfrak{h}}(\omega) \\ &= G(\omega) + \text{sign}(\omega)G(\omega) \\ &= (1 + \text{sign}\,(\omega))\,G(\omega)\cdot \end{aligned} \tag{6.6}$$

In other words, the analytic representation of $g(x)$ can be obtained by discarding its negative frequency components. Moreover, due to the fact that Hilbert transform of a real signal is also real, Eq. 6.3 can be written in Euler form as:

$$g_{\mathfrak{a}}(x) = \alpha(x)e^{i\varphi(x)}, \tag{6.7}$$

where

$$\alpha(x) = \sqrt{g^2(x) + g_{\mathfrak{h}}^2(x)}$$

is the local energy (or local amplitude) and

$$\varphi(x) = \arctan\frac{g_{\mathfrak{h}}(x)}{g(x)}$$

is the local phase. In practice, the analytic signal, and hence the local phase and energy, are computed for a band-passed version of $g(x)$ in order to improve the frequency localization and make the result invariant to the signal energy (by removing the DC) [Bouk 04]. In addition, the band-pass filter is usually designed as an even filter $e(x)$ because it has a constant phase, and thus, it does not change the phase information of the original signal $g(x)$ [Bouk 04]. Based on these justifications, the analytic signal is computed in practical applications by the following equation:

$$\hat{g}_{\mathfrak{a}}(x) = g(x) * e(x) + i\mathcal{H}(g(x) * e(x))\cdot \tag{6.8}$$

According to the convolution property [Poul 10] of Hilbert transform:

$$\begin{aligned} \hat{g}_{\mathfrak{a}}(x) &= g(x) * e(x) + ig(x) * \mathcal{H}(e(x)) \\ &= g(x) * (e(x) + ie_{\mathfrak{h}}(x)) \\ &= g(x) * e_{\mathfrak{a}}(x)\cdot \end{aligned} \tag{6.9}$$

In other words, finding the analytic representation of the signal filtered by $e(x)$ is equivalent to convolving this signal with a *quadrature filter* $e_{\mathfrak{a}}(x)$ which is the analytic representation of $e(x)$. Furthermore, the Hilbert transform of a real even function is a real odd function $o(x)$. Accordingly, one can write:

$$\hat{g}_{\mathfrak{a}}(x) = g(x) * (e(x) + io(x))\cdot \tag{6.10}$$

Consequently, the local energy and phase are computed in practice as:

$$\hat{\alpha}(x) = \sqrt{(g(x) * e(x))^2 + (g(x) * o(x))^2} \tag{6.11}$$

$$\hat{\varphi}(x) = \arctan\frac{g(x) * o(x)}{g(x) * e(x)}\cdot \tag{6.12}$$

Several band-pass filters have been considered in the literature: Gabor, Gaussian derivatives, difference of Gaussians, and others. A thorough discussion about the choice of quadrature filters can be found in [Bouk 04].

6.3.2 Two-dimensional case:

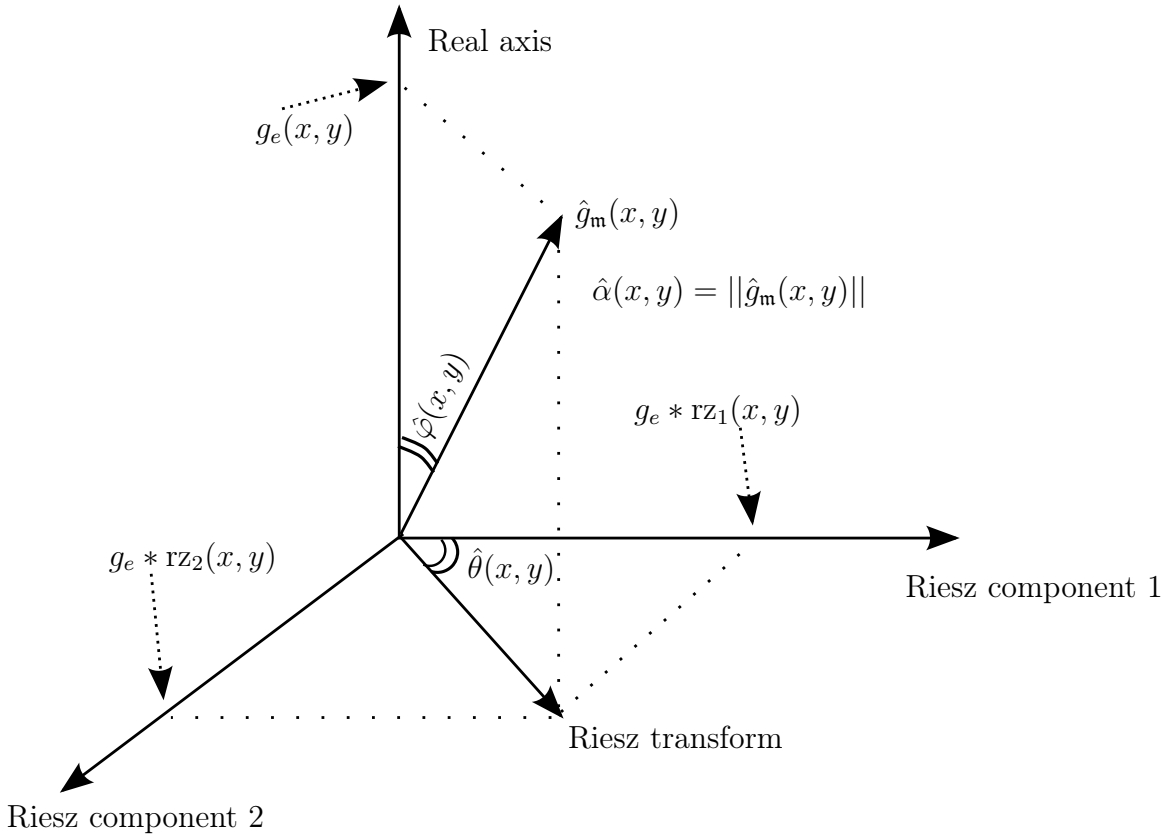


Figure 6.3: Illustration of the monogenic representation of a two-dimensional signal $g(x,y)$ at an arbitrary spatial point. In practice, a band-passed version of $g(x,y)$ is used: $g_e(x,y) = g(x,y) * e(x,y)$. The monogenic signal value $\hat{g}_{\mathfrak{m}}(x,y)$ can be seen as a quaternion whose real part is the signal value $g_e(x,y)$ and its vector part is the Riesz transform of g_e at (x,y). The monogenic features describe this quaternion as follows: local amplitude $\hat{\alpha}(x,y)$ is the magnitude of the quaternion, local orientation $\hat{\theta}(x,y)$ describes the direction of the quaternion's vector part, and local phase $\hat{\varphi}(x,y)$ describes the ratio between the magnitude of the quaternion's vector part and the quaternion's real part.

The Riesz transform generalizes the Hilbert transform for n-dimensional signals [Stei 70]:

$$\boldsymbol{\mathcal{R}}(g(\mathbf{x})) := (\mathcal{R}_1(g(\mathbf{x})), \ldots, \mathcal{R}_n(g(\mathbf{x})))^T \tag{6.13}$$

$$\mathcal{R}_l(g(\mathbf{x})) := \mathrm{rz}_l(\mathbf{x}) * g(\mathbf{x}), l = 1, \ldots, n \tag{6.14}$$

$$\mathrm{rz}_l(\mathbf{x}) := \frac{\Gamma((n+1)/2)}{\pi^{(n+1)/2}} \frac{x_l}{\|\mathbf{x}\|^{n+1}} \tag{6.15}$$

where Γ is the Gamma function [Poul 10], and $\mathbf{x} = (x_1, \ldots, x_n)$. Eq. 6.15 can be written in Fourier domain as:

$$\mathrm{RZ}_l(\mathbf{u}) = i\frac{u_l}{\|\mathbf{u}\|} \tag{6.16}$$

where $\mathbf{u} = (u_1, \ldots, u_n)$ is the n-dimensional frequency vector. For $n = 1$, this transfer function expresses the Hilbert transform[1].

Without loss of generality, the monogenic signal is defined for two-dimensional signals $g(x, y)$ as:

$$\begin{aligned} g_{\mathfrak{m}}(x,y) &= g(x,y) + i\mathrm{rz}_1(x,y) * g(x,y) \\ &\quad + \acute{i}\mathrm{rz}_2(x,y) * g(x,y)\cdot \end{aligned} \tag{6.17}$$

The filters rz_1 and rz_2 are given by Eq. 6.15. $g_{\mathfrak{m}}(x, y)$ is defined in a quaternion space [Hami 44] whose imaginary units are i, $\acute{i}$, and $\acute{\acute{i}}$ ($i^2 = \acute{i}^2 = \acute{\acute{i}}^2 = -1$) and the $\acute{\acute{i}}$-component is zero. One can see the monogenic signal of the two-dimensional function $g(x, y)$ as a quaternion-valued function whose real part is the signal itself and whose vector part is the Riesz transform of the signal.

Similar to the one-dimensional case, in practice, a band-passed version of the signal is used:

$$\begin{aligned} \hat{g}_{\mathfrak{m}}(x,y) &= e(x,y) * g(x,y) + i\mathrm{rz}_1(x,y) * e(x,y) * g(x,y) \\ &\quad + \acute{i}\mathrm{rz}_2(x,y) * e(x,y) * g(x,y)\cdot \end{aligned} \tag{6.18}$$

This can be reformulated as follows:

$$\begin{aligned} \hat{g}_{\mathfrak{m}}(x,y) &= g(x,y) * (e(x,y) + i\mathrm{rz}_1(x,y) * e(x,y) \\ &\quad + \acute{i}\mathrm{rz}_2(x,y) * e(x,y)) \\ &= g(x,y) * e_{\mathfrak{m}}(x,y)\cdot \end{aligned} \tag{6.19}$$

This equation is similar to Eq. 6.9 in the one-dimensional case. It states that computing the monogenic representation of $g(x, y)$ filtered with $e(x, y)$ is equivalent to convolving the signal with a 2D quadrature filter given by the monogenic representation of $e(x, y)$.

Local energy is defined as the magnitude of the monogenic quaternion:

$$\hat{\alpha}(x,y) = \sqrt{(g_e)^2 + (g_e * \mathrm{rz}_1)^2 + (g_e * \mathrm{rz}_2)^2}, \tag{6.20}$$

where $g_e := g(x, y) * e(x, y)$. The specification of the domain (x, y) was omitted in order to simplify the notation. The local phase is defined as the angle between the vector part and the real part of the monogenic quaternion:

$$\hat{\varphi}(x,y) = \arctan\frac{\sqrt{(g_e * \mathrm{rz}_1)^2 + (g_e * \mathrm{rz}_2)^2}}{g_e}\cdot \tag{6.21}$$

Unlike the one-dimensional case, the Riesz transform is a vector and it thus has a direction in the domain of $g_e(x, y)$:

$$\hat{\theta}(x,y) = \arctan\frac{g_e * \mathrm{rz}_2}{g_e * \mathrm{rz}_1}\cdot \tag{6.22}$$

[1]There is a minus sign difference due to the incompatibility of definitions between different authors [Poul 10, Fels 01]. This incompatibility, however, is irrelevant for the discriminative power.

The angle $\hat{\theta}$ is called *local orientation* in the terminology of monogenic representation. Looking at the Riesz transform kernels given in Eq. 6.15, one can see that they partially resemble derivative computation [Koth 05]. Local orientation can be thus understood as the direction of maximal *change*. This change is, however, defined in a way given by Riesz kernels rather than traditional gradient kernels.

In short, for a two-dimensional signal, the monogenic signal can be represented by a quaternion at each domain point (x, y). Since the vector part of this quaternion is two-dimensional, the quaternion can be represented in three dimensions. This representation is illustrated in Figure 6.3.

6.4 Approximating TIE's Solution Using Monogenic Signal

Typically, in order to compute the monogenic features of an image $I(x, y)$, one needs to set g_e in Eq. 6.20, Eq. 6.21, and Eq. 6.22 to $I(x, y) * e(x, y)$, where $e(x, y)$ is an even band-pass filter.

According to [Ali 10], it is possible to use the monogenic signal framework to approximate the solution of Eq. 6.1 under two conditions: Firstly, the derivative image, i. e. the left-hand side of Eq. 6.1, is used as an input of the monogenic framework instead of the image itself. Secondly, a low-pass filter $\mathrm{IL}(x, y)$ which resembles the inverse Laplacian, is used in the monogenic framework instead of the typically-used band-pass filter. More specifically, one needs to set g_e in Eq. 6.20, Eq. 6.21, and Eq. 6.22 to $\frac{\partial I}{\partial z}(x, y) * \mathrm{IL}(x, y)$ instead of $I(x, y) * e(x, y)$.

Under this setup, the low-pass monogenic local phase is given by the following equation:

$$\hat{\varphi}^{\mathrm{lowpass}}(x, y) = \arctan \frac{\sqrt{(\frac{\partial I}{\partial z} * \mathrm{IL} * \mathrm{rz}_1)^2 + (\frac{\partial I}{\partial z} * \mathrm{IL} * \mathrm{rz}_2)^2}}{\frac{\partial I}{\partial z} * \mathrm{IL}}. \tag{6.23}$$

Moreover, the low-pass monogenic local amplitude is given as:

$$\hat{\alpha}^{\mathrm{lowpass}}(x, y) = \sqrt{(\frac{\partial I}{\partial z} * \mathrm{IL})^2 + (\frac{\partial I}{\partial z} * \mathrm{IL} * \mathrm{rz}_1)^2 + (\frac{\partial I}{\partial z} * \mathrm{IL} * \mathrm{rz}_2)^2}. \tag{6.24}$$

The specification of the domain (x, y) was again omitted from Eq. 6.23 and Eq. 6.24 in order to simplify the notation. The employed low-pass filter in [Ali 10] was a Mellor-Brady filter [Mell 05] given by the following equation in the spatial domain:

$$\Omega(x, y, \nu_1, \nu_2) := \frac{1}{(x^2 + y^2)^{\frac{\nu_1 + \nu_2}{2}}} - \frac{1}{(x^2 + y^2)^{\frac{\nu_1 - \nu_2}{2}}}, \tag{6.25}$$

where ν_1 and ν_2 are the filter parameters. Depending on these parameters, Ω behaves either as a low-pass or a band-pass filter. This tunability facilitates suppressing the low-frequency noise which usually perturbs TIE-solution images. For approximating the TIE's solution, a low-pass filter which resembles the inverse Laplacian was employed corresponding to $\nu_1 = \nu_2 = 0.25$. Therefore, IL in Eq. 6.23 and Eq. 6.24 was set to $\Omega(x, y, 0.25, 0.25)$.

Physical phase vs. local phase: Eq. 6.23 computes the monogenic local phase of $\frac{\partial I}{\partial z} * \mathrm{IL}$. The connection to TIE becomes clear in the special case when I can be considered uniform in the lateral plane (cf. Section 2.6). Under this condition, an expression simpler than Eq. 6.2 for physical light phase ϕ can be directly derived from Eq. 2.23:

$$\check{\phi} = C_0 \nabla^{-2} \left(\frac{\partial I}{\partial z} \right), \tag{6.26}$$

where C_0 is a constant related to wavelength and the uniform intensity value. Therefore, Eq. 6.23 computes the local phase of an approximation of physical light phase.

Note about terminology: in this thesis, all cell images of local phase and local amplitude were generated under the specific setup described above. Accordingly, in the remaining text, the term *local phase* refers to Eq. 6.23 while *local energy* and *local amplitude* refer to Eq. 6.24. On the other hand, *physical phase* and *TIE solution* both refer to Eq. 6.2.

6.5 Cell Detection Pipeline Customized for Joint Learning

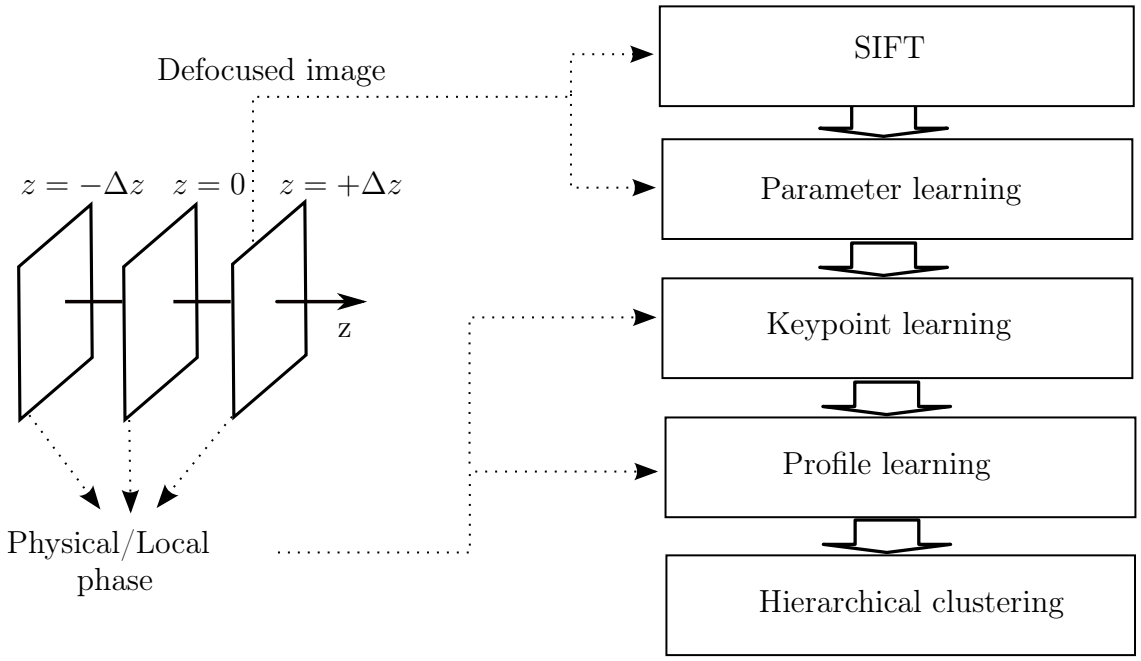


Figure 6.4: Extending the cell detection pipeline of Chapter 5 using phase-based features for improving joint learning

We here extend the pipeline of our supervised approach presented in Chapter 5. Similar to Chapter 5, we use a defocused image for keypoint extraction and parameter learning. However, in contrast to Chapter 5, we extract the keypoint features and the profile features from a TIE solution or local phase image instead of a defocused image. Figure 6.4 clarifies the structure of the proposed pipeline.

A difference of two defocused images at distances $\pm\Delta z$ is used as an estimation of axial derivative. Local phase is then computed directly using Eq. 6.23. The exact values of Δz for each cell line are given in Section 6.6. The estimated axial derivative and the image at-focus are used to compute the TIE solution using Eq. 6.2. Inversion of the Laplacian in Eq. 6.2 is performed by applying a Fourier-based method [Volk 02]. Usually, the resulting solution is contaminated with a low-frequency bias field. The latter is estimated using a thin-plate smoothing spline approach and subtracted from the TIE solution.

6.6 Generating TIE and Monogenic Images

In this section, we describe the use of images in the real cell lines of Table 4.1 for generating low-pass monogenic outputs and TIE-solution images. Simulated cell lines were not used because defocusing was not included in the applied simulation model. As described in Table 4.1, we use three real cell lines containing together 16 image sets. Each image set is composed of a negatively defocused image at defocus distance $-\Delta z$ (Figure 6.5a), an image at focus (Figure 6.5b), and a positively defocused image at defocus distance $+\Delta z$ (Figure 6.5c). The used Δz values are, as mentioned earlier in Section 4.1.2, 30 μm for the adherent cell lines and 15 μm for the suspension cell line. The software package SePhaCe [Ali 12] was utilized to generate a local energy image (Figure 6.5d), a local phase image (Figure 6.5e), and a TIE-solution image for each image set. SePhaCe was also used for the bias correction step (cf. Section 6.5). A thin-plate smoothing spline was estimated for each TIE-solution image (1280×960 pixels) over a grid of 50×50 points. Figure 6.5f exemplifies the TIE solution after subtracting the bias field.

6.7 Evaluation

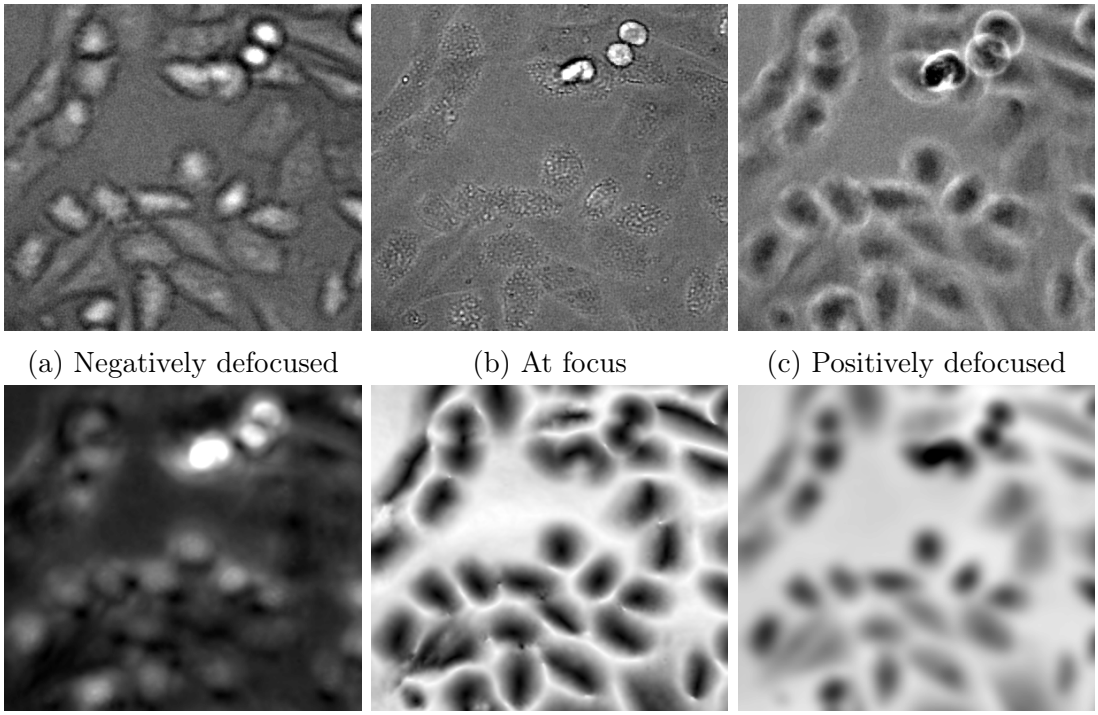
This section is composed of two main parts: in Section 6.7.1, the discriminative power of low-pass monogenic outputs for the cell/background separation problem is compared to the discriminative power of defocused images. In Section 6.7.2, the phase-based cell detection pipeline proposed in Section 6.5 is evaluated and compared to the pipeline presented in Chapter 5.

6.7.1 Evaluation of the Discriminative Power of Low-pass Monogenic Signal for Cell/Background Separation

6.7.1.1 Cell/Background Pixelwise Classification

We employ machine learning to investigate the discriminative power of the local phase as defined in Eq. 6.23 in the cell/background separation problem. Obviously, it is possible to measure the difference in the discriminative power of two features by learning a classifier for each of them and then comparing the test errors.

As classifier features, patches of size 5×5 pixels are used. Cell areas in our data are considerably larger than the area of the chosen patch. The advantage of using



(a) Negatively defocused (b) At focus (c) Positively defocused

(d) Local energy (e) Local phase (f) TIE solution after bias correction

Figure 6.5: Patches extracted from an L929 image set. The histogram of each patch was linearly stretched in the range $[0, 255]$ for clarity.

a small patch size is reducing the sensitivity of the extracted feature vectors to the variability of cell orientation. Analysis of the effect of patch size is conducted in Section 6.7.1.5.

As a classifier model, we use SVM and RF (random forest). Two kernels were utilized for the SVM: the radial basis function (RBF) and the linear kernel. The SVM cost parameter and the RBF width parameter were set to the default values of LibSVM [Chan 11]. The number of trees N_{Tr} in the random forest and the number of randomly selected variables N_{Rand} at each node were set after [Khos 07] to 500 and $N_F/5$, respectively, where N_F is the number of features. The data was z-scored for the SVM, while it was used without normalization for the random forest.

One can extract a patch at each pixel from all images. However, this is computationally expensive. Therefore, only NP patches are randomly sampled from each training/testing image. Unless otherwise specified, NP is set to 100. In order to achieve balanced learning, one-half of the NP patches are sampled from background while the other half sampled from cells. The class of each patch is obtained from the ground-truth masks. The ground truth for inhomogeneous patches, i. e. patches which contain both labels (usually near cell boundary), is not reliably known. Consequently, unless otherwise stated, these patches are discarded.

	Defocused (%)	At-focus (%)	Phase (%)	Energy (%)
Linear SVM	67.7 ± 0.5	51.2 ± 1.0	**82.4** ± 1.1	64.2 ± 1.3
RBF SVM	68.2 ± 1.5	56.2 ± 1.5	**81.7** ± 0.9	64.5 ± 1.9
RF	68.1 ± 1.1	54.9 ± 0.9	**80.3** ± 1.1	60.6 ± 1.3

Table 6.1: L929: comparison between local phase, local energy, at-focus signal, and defocused signal using the cell/background classification rate

	Defocused (%)	At-focus (%)	Phase (%)	Energy (%)
Linear SVM	82.5 ± 1.2	59.6 ± 1.2	**94.9** ± 0.7	88.5 ± 0.5
RBF SVM	84.1 ± 0.8	73.8 ± 2.0	**94.9** ± 0.6	88.8 ± 0.8
RF	87.4 ± 1.3	70.8 ± 2.9	**94.6** ± 0.7	88.4 ± 1.1

Table 6.2: Sf21: comparison between local phase, local energy, at-focus signal, and defocused signal using the cell/background classification rate

	Defocused (%)	At-focus (%)	Phase (%)	Energy (%)
Linear SVM	60.9 ± 1.2	49.8 ± 0.8	**68.5** ± 1.7	57.4 ± 1.9
RBF SVM	61.7 ± 0.9	52.4 ± 0.6	**68.0** ± 2.4	55.7 ± 1.3
RF	61.1 ± 1.5	52.5 ± 0.5	**63.7** ± 1.5	54.8 ± 0.9

Table 6.3: CHO: comparison between local phase, local energy, at-focus signal, and defocused signal using the cell/background classification rate

6.7.1.2 Comparison Between Local Phase, Local Energy, At-focus Signal, and Defocused Signal

One of the five defocused L929 images I_l was used to train three classifiers: linear SVM, RBF SVM, and random forest. The learned models were then applied on the other defocused images in the same cell line and the average classification rate CR_l over these test images was computed. This was repeated for each defocused L929 image, i. e. for each l value, and the mean of the CR_l values was obtained.

The previous experiment was repeated 10 times with one mean classification rate obtained from each repetition. The mean and the standard deviation of all these mean classification rates can be seen in the first column of Table 6.1. The second, third, and fourth column of the same table show the results when the same process was applied on the at-focus, local phase, and local energy images, respectively. Table 6.2 and Table 6.3 show the results of the same procedure applied on Sf21 and CHO.

Tables 6.1, 6.2, and 6.3 reveal that the four features can be sorted by increasing discriminative power as follows: at-focus signal, local energy, defocused signal, local phase. The only exception for this order is that local energy is more discriminative than defocused signal for suspended cells (Sf21).

6.7.1.3 Comparison Between the Input Space and the Output Space of the Monogenic Signal

In this section, we assess the use of the two monogenic outputs together for cell/background classification and compare it with the joint use of the two monogenic inputs. In this

	At-focus and defocused (%)	Phase and energy (%)
Linear SVM	72.8 ± 1.2	81.6 ± 1.1
RBF SVM	71.4 ± 1.7	81.5 ± 0.8
RF	73.2 ± 1.5	79.1 ± 1.9

Table 6.4: L929: comparison between the input space and the output space of the monogenic signal using the cell/background classification rate

	At-focus and defocused (%)	Phase and energy (%)
Linear SVM	84.9 ± 1.0	96.2 ± 0.6
RBF SVM	86.1 ± 1.5	97.1 ± 0.6
RF	87.2 ± 2.4	95.7 ± 0.6

Table 6.5: Sf21: comparison between the input space and the output space of the monogenic signal using the cell/background classification rate

case, at a given pixel, a patch from a local phase image and another patch at the same pixel position from its corresponding local energy image are extracted. The values of the two patches are then concatenated. Therefore, the dimensionality of the resulting feature space is $25 + 25 = 50$. The discrimination power of this feature space was compared with another 50-dimensional feature space: the monogenic input space. The latter is formed by using an at-focus image and its corresponding defocused image together for patch extraction instead of the local phase and energy images.

The first column of Table 6.4 shows the cell/background classification rate on L929 when both an at-focus image and a defocused image are used together to train the classifiers. The second column shows the classification rate when both local phase and local energy are used to train the classifiers. The same can be seen in Table 6.5 for Sf21 and Table 6.6 for CHO. The classification rate was estimated in a way similar to the evaluation procedure in Section 6.7.1.2. However, compared to Section 6.7.1.2, the dimensionality of the feature space is 50 instead of 25. Tables 6.4, 6.5, and 6.6 reveal that the compound signal of local phase and local energy is more discriminative than the compound signal of an at-focus image and a defocused image.

	At-focus and defocused (%)	Phase and energy (%)
Linear SVM	58.0 ± 1.2	67.6 ± 1.4
RBF SVM	55.7 ± 0.6	67.0 ± 1.7
RF	57.1 ± 1.6	65.0 ± 1.6

Table 6.6: CHO: comparison between the input space and the output space of the monogenic signal using the cell/background classification rate

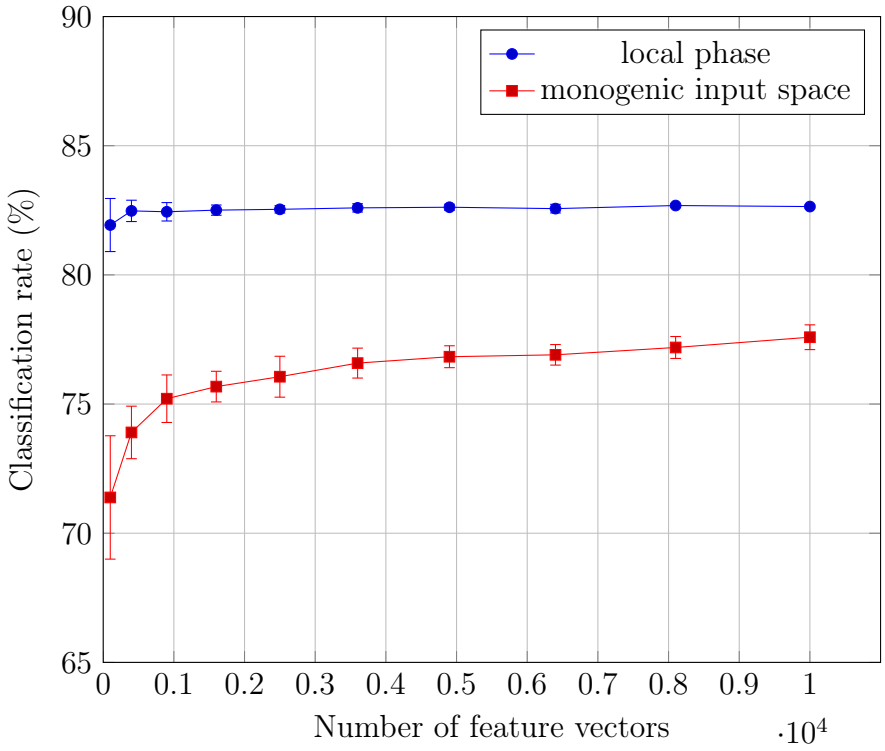


Figure 6.6: Comparison between learning curves of a RBF SVM on L929 using two feature spaces: 1) local phase 2) monogenic input space

6.7.1.4 Comparison Between Local Phase and the Input Space of the Monogenic Signal

Figure 6.6 shows a comparison between the learning curve of a RBF SVM trained using both a defocused image and an at-focus image compared to the learning curve of the same classifier model trained using a local phase image. For each point in the curve, i. e. for each number of patches $\mathrm{NP}_q = 100q^2, q = 1..10$, the classification rate was estimated in a cross-validation loop similar to the loop described in Section 6.7.1.2. The learning curve shows that a local phase image is more discriminative than the two images which were used to generate it even when more training data is incorporated in order to compensate for the increased dimensionality of the feature space.

6.7.1.5 Patch Size Analysis

All experiments in our evaluation were performed so far with 5×5 sized patches. In this section, we investigate other patch sizes. Table 6.7 shows the classification rate of a RBF SVM classifier on L929 employing the same evaluation scheme described in Section 6.7.1.2 but using 23×23, 33×33, and 43×43 sized patches. In order to give the reader a feeling about the ratio between these patch dimensions and cell dimensions, we point out that the length of minor axis of the L929 cells in our data is 32.95 ± 10.72 pixels.

	Defocused (%)	Phase (%)
5×5	68.2 ± 1.5	81.7 ± 0.9
23×23	63.4 ± 1.3	89.3 ± 0.8
33×33	56.5 ± 1.8	88.8 ± 0.6
43×43	52.2 ± 1.9	86.7 ± 1.7

Table 6.7: The effect of patch size on the cell/background classification rate. The cell line is L929, the classifier model is RBF SVM, and the number of patches per training/testing image is 100.

	Defocused (%)	Phase (%)
5×5	68.6 ± 0.4	82.4 ± 0.3
23×23	65.6 ± 0.4	89.8 ± 0.4
33×33	58.9 ± 0.7	87.9 ± 0.4
43×43	53.8 ± 0.6	82.4 ± 1.0

Table 6.8: The effect of patch size on the cell/background classification rate. The cell line is L929, the classifier model is RBF SVM, and the number of patches per training/testing image is 900.

Increasing the patch size increases the dimensionality of the corresponding feature space, and consequently, the number of samples needed for training. Table 6.8 shows the same experiment reported in Table 6.7, but with 900 random patches per training/testing image instead of 100.

As stated in Section 6.7.1.1, only homogeneous patches are used in training and testing. This is a plausible choice when the patch's area is small compared to cell's area (e. g. 5×5 sized patches). However, with larger patches, this will exclude more cell pixels from the evaluation scheme and hence degrade the generalizability of the derived conclusions. Table 6.9 shows the results when no patches are excluded from the evaluation. In this case, the label of the patch's center is considered.

Tables 6.7, 6.8, and 6.9 reveal that the superiority of local phase over defocused signal holds for larger patches even when more samples are employed in training. On the other hand, unlike the discriminative power of local phase, the discriminative power of defocused signal benefits from employing near-boundary patches in training.

	Defocused (%)	Phase (%)
5×5	65.2 ± 1.5	76.9 ± 0.7
23×23	66.2 ± 1.3	77.3 ± 1.0
33×33	65.5 ± 1.9	76.9 ± 0.8
43×43	67.7 ± 1.3	76.1 ± 1.3

Table 6.9: The effect of patch size on the cell/background classification rate. The cell line is L929, the classifier model is RBF SVM, and the number of patches per training/testing image is 100. Inhomogeneous patches are included in training and testing.

6.7.2 Evaluation of the Phase-based Cell Detection Pipeline

In this section, we compare the joint learning performance between our supervised approach in Chapter 5 (referred to as original pipeline in this section) and its extended version presented in Section 6.5. In order to evaluate the original pipeline for joint learning of Sf21 and L929, a positively defocused image of each of the two cell lines was randomly chosen. The two images were used to train the system. The trained system was then tested on the rest of the positively defocused images in L929 and Sf21 and a mean F-measure value was obtained. This process was repeated five times with a mean F-measure value obtained from each repetition. The average and the standard deviation of these five mean F-measure values can be seen at the left-hand side of the first row of Table 6.10. Since adherent and suspended cell images have different dynamic ranges, each image was normalized to $[0, 1]$ before being used for training or testing.

The right-hand side of the first row shows F-measure results for separate training. In this case, one image is used for training in each cell line and tested on the other images of the same cell line. This is repeated in a cross-validation loop.

We evaluated the extended pipeline for joint and separate training in the same manner as described above. The second row of Table 6.10 shows the F-measure results when TIE solution is used for keypoint and profile feature extraction. The third row shows the results when local phase images are used for feature extraction. And lastly, the fourth row is dedicated to the case when TIE solution is used for keypoint feature extraction while local phase is used for profile feature extraction.

In Table 6.11, the same experiment was performed but with CHO as adherent cell line instead of L929. It is possible to average the results obtained from Table 6.10 and Table 6.11 and shape the figures in terms of suspended and adherent cell lines. This can be seen in Table 6.12. In this table, one can notice that the F-measure on suspended cells using the original pipeline was reduced from 97.0 in separate training to 86.6 in joint training. When, for instance, TIE solution was employed for keypoint and profile feature extraction, F-measure on suspended cells was recovered to 95.7 while F-measure on adherent cells degraded with a very small amount (from 84.2 to 83.6). In order to draw conclusions more easily, in Table 6.13 we aggregate the results of Table 6.12 in terms of total loss in F-measure, i. e. the loss in F-measure on adherent cells added to the F-measure loss on suspended cells. One can see that the minimum loss in joint training is obtained when TIE solution is used for keypoint feature extraction and local phase is used for profile feature extraction. The other two cases (TIE alone or local phase alone) were a bit inferior to TIE with local phase, but very close to it. On the other hand, in separate training, the difference between all four cases was small. Nevertheless, the extended pipeline achieved a bit higher separate-learning F-measure compared to the original pipeline.

6.8 Discussion

It was empirically shown that the pixelwise cell/background classification yields considerably better results when local phase as obtained in [Ali 10] is used instead of a defocused image. More generally, the feature images can be sorted by increasing

	Joint learning		Separate learning	
	L929	Sf21	L929	Sf21
Original pipeline	85.3 ± 2.1	86.7 ± 2.4	86.5 ± 1.3	97.0 ± 1.2
KFI = PFI = TIE	84.5 ± 2.1	95.7 ± 0.4	87.1 ± 1.7	97.7 ± 0.9
KFI = PFI = local phase	83.6 ± 1.9	95.5 ± 1.5	85.7 ± 1.9	98.2 ± 0.7
KFI = TIE, PFI = local phase	84.2 ± 2.2	96.5 ± 0.6	86.9 ± 1.7	97.8 ± 0.9

Table 6.10: F-measure values of the original and extended approaches for L929 and Sf21. KFI denotes the image used for keypoint feature extraction while PFI denotes the image used for profile feature extraction.

	Joint learning		Separate learning	
	CHO	Sf21	CHO	Sf21
Original pipeline	83.1 ± 4.0	86.5 ± 3.3	84.2 ± 3.4	97.0 ± 1.2
KFI = PFI = TIE	82.7 ± 3.4	95.7 ± 1.4	84.4 ± 2.5	97.7 ± 0.9
KFI = PFI = local phase	81.6 ± 2.7	94.5 ± 1.8	83.9 ± 2.3	98.2 ± 0.7
KFI = TIE, PFI = local phase	83.0 ± 3.6	96.1 ± 1.1	84.1 ± 2.5	97.8 ± 0.9

Table 6.11: F-measure values of the original and extended approaches for CHO and Sf21. KFI denotes the image used for keypoint feature extraction while PFI denotes the image used for profile feature extraction.

discriminative power as follows: at-focus signal, local energy, defocused signal, local phase. The only exception to this order was the superiority of local energy over defocused signal for suspended cells.

In addition, we showed that the monogenic output space is more discriminative than the monogenic input space. This is probably due to the following reason: the monogenic output delivers information about the physical light phase represented in a way which describes signal features. In fact, there is a relation between the signal features, e. g. edges and blobs, and the local phase and energy of this signal. Local energy is high at distinctive signal features while local phase determines the feature type [Morr 86].

In the pixelwise cell/background classification experiments, a local phase image is an output of the low-pass monogenic signal framework with an at-focus image and a defocused image used as inputs. The natural question which arises here is whether using both input images together could deliver the same discriminative power obtained by the local phase image. Due to the difference in dimensionality between the two feature spaces, more samples need to be provided for the higher dimensional feature space in order to achieve a fair comparison. For this reason, the learning curve was utilized to compare local phase with the input of the monogenic signal framework. The results show that by increasing the size of training data, local phase is still more discriminative than the monogenic input.

As stated in [Ali 10], the use of low-pass filters for computing local phase and energy is "against the accepted theory". We think, however, that there is a kind

	Joint learning		Separate learning	
	Adherent	Suspended	Adherent	Suspended
Original pipeline	84.2	86.6	85.3	97.0
KFI = PFI = TIE	83.6	95.7	85.7	97.7
KFI = PFI = local phase	82.6	95.0	84.8	98.2
KFI = TIE, PFI = local phase	83.6	96.3	85.5	97.8

Table 6.12: F-measure values of the original and extended approaches averaged from Table 6.10 and Table 6.11. KFI denotes the image used for keypoint feature extraction while PFI denotes the image used for profile feature extraction.

	Joint learning	Separate learning
Original pipeline	29.2	17.6
KFI = PFI = TIE	20.7	16.5
KFI = PFI = local phase	22.4	17.0
KFI = TIE, PFI = local phase	20.1	16.7

Table 6.13: Total F-measure loss of the original and extended approaches. KFI denotes the image used for keypoint feature extraction while PFI denotes the image used for profile feature extraction.

of DC subtraction implicitly injected as follows: approximating the axial derivative involves subtracting the image at focus from a defocused image. For thin cells, there is almost no information at focus. Therefore, the difference of the two images partially resembles a subtraction of the DC component.

The ground truth was defined by delineating cell borders in the defocused images (cf. Section 4.1). The latter are blurred compared to the at-focus images and defocused cells tend to occupy larger area (cf. Figure 6.5b and Figure 6.5c). Therefore, comparing the accuracy of pixelwise classification between a defocused image and an at-focus image is slightly biased. This bias is small because the random sampling and the exclusion of inhomogeneous patches make the probability of selecting a pixel which belongs to a defocused cell but not to the corresponding at-focus cell very low. In addition, as mentioned in Chapter 1, the superiority of the defocused image over the at-focus image in the cell/background separation is already known in literature, and hence it is not a main concern in the evaluation.

We then employed physical phase and local phase for improving joint learning of adherent and suspended cell detection. It was shown that joint learning capability of the cell detection approach introduced in Chapter 5 can be substantially improved by using a TIE-solution image or local phase image for feature extraction. However, compared to the pipeline presented in Chapter 5, the extended phase-based pipeline presented in this chapter is less general as it is based on the image formation model in bright field microscopy.

One might criticize the evaluation as being done using a fixed defocus distance, i. e. the distance of 30 μm or 15 μm described in Section 6.6. Our dataset does not

contain focus stacks. Therefore, no evaluation of the effect of defocus distance selection on pixelwise classification or joint learning of cell detection was performed. The selection of defocus distance is, however, not arbitrary. The very short distances do not deliver sufficient contrast. On the other hand, very long distances smear the image information due to the excessive blurring by point spread function of the optical system. Therefore, there is an optimal distance which maximizes the contrast. During the image acquisition, we tried to select this optimal distance experimentally. However, this was judged subjectively. Automatic methods to choose this distance objectively need thus to be developed in future. Compared to published phase retrieval results for cell segmentation [Curl 04, Ali 12], our phase images look blurred. Therefore, shorter distances are needed for border delineation. However, our concern is to maximize contrast for cell detection applications rather than cell segmentation. Consequently, using the aforementioned contrast-blurring trade-off principle for the defocus distance selection sounds plausible.

Chapter 7

Unsupervised Cell Detection

Considerable parts of this chapter were already published in:
F. Mualla, S. Schöll, B. Sommerfeldt, A. Maier, S. Steidl, R. Buchholz, and J. Hornegger. "Unsupervised unstained cell detection by SIFT keypoint clustering and self-labeling algorithm". In: P. Golland, N. Hata, C. Barillot, J. Hornegger, and R. Howe, Eds., *Medical Image Computing and Computer-Assisted Intervention MICCAI 2014*, pp. 377–384, Springer International Publishing, Boston, MA, USA, September 2014.

7.1 Motivation

The cell detection approaches discussed so far depend on supervised learning. The latter transfers part of its inductive bias to the training data which makes the approach adaptable by simply changing the training set. This has the advantage that it can model very complicated situations and provide reliable results as long as the training set is representative. On the other hand, its drawback is that it requires labeled ground truth. In many cases, the users of cell image analysis software would sacrifice some detection accuracy in favor of having a labeling-free system. This preference becomes more serious when the system has to be trained for each new cell line.

In Chapter 5, we presented a supervised algorithm for cell detection in which strong emphasis was placed on reliability and robustness. In this chapter, no claim is made that it is possible to obtain the same degree of robustness without supervision. We show, however, that part of the knowledge needed to detect cells in an image can be learned from this image based on unsupervised learning. Technically speaking, we also employ supervised learning, but with ground truth learned automatically from the input image. Moreover, we apply the algorithm on images obtained by both bright field microscopy and phase contrast microscopy which form together a very appropriate choice for the evaluation of unstained cell detection.

The rest of the chapter is organized as follows: Section 7.2 describes the proposed algorithm. The results are given in Section 7.3 which are discussed along with final conclusions in Section 7.4.

7.2 Cell Detection by Keypoint Clustering and Self-labeling Algorithm

The proposed unsupervised approach is composed of a sequence of steps. These steps are described in order of application in the following sections.

7.2.1 Keypoint Extraction

The algorithm starts by extracting SIFT keypoints of the input image I. These keypoints are not thresholded using the PCR or the $\mathrm{DoG}_{\gamma 1}$ values. In other words, all detected SIFT keypoints of all strength and anisotropy values are considered at this step.

7.2.2 Blob Type Detection

As mentioned in Section 5.3, the blob type is either black-on-white (+1), or white-on-black (-1). We compute the blob type in this unsupervised approach based on the following equation:

$$\mathrm{BT}_{\mathrm{unsup}} = \mathrm{sign}\left(\frac{\sum_{j=1}^{N_{kp}} \eta(\mathbf{p}_j)\,|\mathrm{DoG}_{\gamma 1}(\mathbf{p}_j)|\,\mathrm{HS}(\mathrm{DoG}_{\gamma 1}(\mathbf{p}_j))}{\sum_{j=1}^{N_{kp}} \eta(\mathbf{p}_j)\,|\mathrm{DoG}_{\gamma 1}(\mathbf{p}_j)|} - \frac{1}{2}\right), \tag{7.1}$$

where $\mathbf{p}_j$, $j = 1..N_{kp}$ are the extracted keypoints in the considered image, N_{kp} is their number, HS is the Heaviside step function (cf. Eq. 5.2), and η is given by:

$$\eta(\mathbf{p}_j) = \frac{\sigma(\mathbf{p}_j)}{\mathrm{PCR}(\mathbf{p}_j)}. \tag{7.2}$$

There are two differences between Eq. 7.1 in the unsupervised approach presented here and Eq. 5.1 in the supervised approach presented in Chapter 5:

1. In the supervised approach, *cell* keypoints in the entire *training dataset* are involved in learning the blob type. As this is not possible when ground truth is not available, in the unsupervised case, we use *all* keypoints extracted from the *input image*.

2. The weight of each $\mathrm{DoG}_{\gamma 1}$ value is slightly changed: instead of using only keypoint scale to weight the contribution of each keypoint's $\mathrm{DoG}_{\gamma 1}$ value, we additionally employ the PCR giving elongated keypoints less importance.

7.2.3 Scale Adaptive Smoothing

The image I is smoothed with a Gaussian kernel whose standard deviation is the mean keypoint scale. The latter is computed using the following equation:

$$\bar{\sigma}_{\mathrm{unsup}} = \frac{\sum_{j=1}^{N'_{kp}} |\mathrm{DoG}_{\gamma 1}(\mathbf{p}_j)|\,\sigma(\mathbf{p}_j)}{\sum_{j=1}^{N'_{kp}} |\mathrm{DoG}_{\gamma 1}(\mathbf{p}_j)|} \tag{7.3}$$

where N'_{kp} is the number of keypoints resulting from the step described in Section 7.2.2, i. e. only one blob type is considered. This is similar to the SAS in the supervised approach (cf. Section 5.4.3) except that we use here a weighted average instead of a simple arithmetic average. The smoothed image $I_{\bar{\sigma}_{\text{unsup}}}$ is saved for further processing.

7.2.4 Second Keypoint Extraction

The step described in Section 7.2.1 is applied on the smoothed image $I_{\bar{\sigma}_{\text{unsup}}}$ and the keypoints which conform to the previously computed BT_{unsup} are considered while the others are discarded. The goal of this step is to adapt the SIFT Gaussian pyramid according to the mean structure size in the input image. The relation between the resulting SIFT pyramid and the original one, i. e. the pyramid resulting from applying SIFT on I, is clarified in the text which follows. Based on Eq. 3.16, the sequence of discrete variance values in a typical SIFT GSS can be given according to the following equation:

$$\begin{aligned}\sigma^2(l) &= \left(\sigma_0 2^{\frac{l}{\mathfrak{S}}}\right)^2 \\ &= \sigma_0^2 2^{\frac{2l}{\mathfrak{S}}}, l = 0 \cdots \mathfrak{S} \cdot \mathfrak{O} - 1 \cdot\end{aligned} \tag{7.4}$$

The reader is referred to Section 3.1.4 for the definitions of constants and variables in Eq. 7.4. Convolving an image consecutively with two Gaussian kernels of variances τ_1 and τ_2 is equivalent to convolving this image with a single kernel of variance $\tau_1 + \tau_2$. Consequently, applying SIFT on the smoothed image $I_{\bar{\sigma}_{\text{unsup}}}$ is equivalent to applying SIFT on the original image I, but with the following discrete variance values for the SIFT GSS:

$$\check{\sigma}^2(l) = \bar{\sigma}^2_{\text{unsup}} + \sigma_0^2 2^{\frac{2l}{\mathfrak{S}}}, l = 0 \cdots \mathfrak{S} \cdot \mathfrak{O} - 1 \cdot \tag{7.5}$$

The discretized σ values of this GSS are thus given as:

$$\check{\sigma}(l) = \sqrt{\bar{\sigma}^2_{\text{unsup}} + \sigma_0^2 2^{\frac{2l}{\mathfrak{S}}}}, l = 0 \cdots \mathfrak{S} \cdot \mathfrak{O} - 1 \cdot \tag{7.6}$$

For instance, the first level in the GSS pyramid of $I_{\bar{\sigma}_{\text{unsup}}}$ is $\sqrt{\bar{\sigma}^2_{\text{unsup}} + \sigma_0^2}$ compared to σ_0 in the GSS pyramid of I.

7.2.5 Cell/background Keypoint Clustering

At this step, the keypoints are clustered into one of two categories: cells and background. $\Bbbk$-medians clustering is applied with $\Bbbk = 2$. $\Bbbk$-medians is a clustering algorithm with an objective function which minimizes intra-cluster $\ell 1$-norm. It is very similar to the more famous $\Bbbk$-means algorithm which minimizes the $\ell 2$-norm inside each cluster. In both variations, i. e. $\Bbbk$-medians and $\Bbbk$-means, a local optimum of the objective function is found by iteratively computing the center (a median in $\Bbbk$-medians and a mean in $\Bbbk$-means) of each cluster and then assigning feature vectors to cluster centers. The local optima are dependent on initialization, and hence, this iterative procedure needs to be properly initialized.

For initializing the cell/background clustering, we employ one-dimensional Otsu thresholding on the $\mathrm{DoG}_{\gamma 1}$ values of the keypoints and the two resulting clusters are used to start the iteration. The features are modality-specific:

- For bright field microscopy, at each keypoint $\mathbf{p}_j$, we employ $\mathrm{DoG}_{\gamma 1}(\mathbf{p}_j)$ and smoothed image intensity $I_{\bar{\sigma}_{\mathrm{unsup}}}(\mathbf{p}_j)$ as features.
- For phase contrast microscopy, we use $\mathrm{DoG}_{\gamma 1}(\mathbf{p}_j)$ and $\mathrm{VMap}'_I(\mathbf{p}_j, 2\bar{\sigma}_{\mathrm{unsup}})$. The latter is the local variance of the original image I within a square neighborhood centered at $\mathbf{p}_j$ with a side-length equal to $2\bar{\sigma}_{\mathrm{unsup}}$ (up to an integer approximation). It is similar to the variance map in Section 5.3.1, except that the window size is given here in terms of $\bar{\sigma}_{\mathrm{unsup}}$ instead of $\sigma(\mathbf{p}_j)$.

The features are normalized to $[0, 1]$ so that they contribute equally to the $\ell 1$-norm. After termination, the keypoints which belong to the background cluster are discarded.

7.2.6 Cell/Cell Keypoint Clustering

The goal of this step is to cluster the cell keypoints resulting from the previous step (Section 7.2.5) into N_{clust} clusters where two keypoints belong to the same cluster if and only if they belong to the same cell. N_{clust} is not known a priori. In order to achieve this goal, similar to Chapter 5, a classifier which ranks each pair of keypoints as belonging to the same cell or not is required. In contrast to Chapter 5, however, we here propose to learn this classifier from the input image using a self-labeling algorithm instead of manually-labeled ground truth. Informally speaking, the algorithm trains a profile classifier on *extreme* cases (for which ground truth labels can be assumed) and applies the resulting classifier on *intermediate* cases. This is achieved as follows:

1. Consider $\mathfrak{A}$ to be a set of keypoint pairs defined as:

$$(\mathbf{p}_j, \mathbf{p}_l) \in \mathfrak{A} \Leftrightarrow \|\mathbf{p}_j - \mathbf{p}_l\|_2 \geq \varpi,$$

   where

$$\varpi = \rho \cdot \bar{\sigma}_{\mathrm{unsup}},$$

   and ρ is a constant. ϖ must be larger than the maximum cell length in pixels. Due to the use of SIFT, *safe* values for ρ can be set easily regardless of the image resolution or cell type. We set it to 10 in our experiments, that is to say every line segment between two cell keypoints which is longer than $10\bar{\sigma}_{\mathrm{unsup}}$ can be assumed to be between two different cells.

2. Randomly choose N_a elements, i. e. keypoint pairs, from $\mathfrak{A}$. Label each of them as *cross* which means that the two corresponding keypoints belong to two different cells and the profile between them is thus a cross profile.

3. Randomly choose N_b keypoints from the set of cell keypoints and form the set $\mathfrak{B}$. In this random sampling, the probability of selecting a keypoint for $\mathfrak{B}$ is, by construction, proportional to its scale, so as to lessen keypoints which indicate very small structures. Both N_a and N_b were set to 100 in our experiments.

4. For each element $\mathbf{p}_j$ in $\mathfrak{B}$, choose a random orientation $\dot{\vartheta}_j$ and construct the point:

$$\tilde{\mathbf{p}}_j = \mathbf{p}_j + \Big(\sigma(\mathbf{p}_j)\cos(\dot{\vartheta}_j), \sigma(\mathbf{p}_j)\sin(\dot{\vartheta}_j)\Big)\,.$$

   The resulting point $\tilde{\mathbf{p}}_j$ is located at a distance $\sigma(\mathbf{p}_j)$ from $\mathbf{p}_j$. Label each pair $(\mathbf{p}_j,\ \tilde{\mathbf{p}}_j)$ as *inner* which means that the two corresponding points belong to the same cell. This is motivated by the intuition that a single cell keypoint is very unlikely to span more than one cell because it represents a structure inside a cell. The labels obtained by this step and by step 2 are illustrated in Figure 7.1 (b).

5. For each *inner*/*cross* pair $(\mathbf{p}_j^*, \mathbf{p}_l^*)$, extract the following feature:

$$V_{jl} = I_{\bar{\sigma}_{\text{unsup}}}(\mathbf{p}_j^*) - 2\ \text{extremum}_{jl} + I_{\bar{\sigma}_{\text{unsup}}}(\mathbf{p}_l^*),$$

   where extremum_{jl} is, by definition, either the maximum (when $\text{BT}_{\text{unsup}} = +1$) or the minimum ($\text{BT}_{\text{unsup}} = -1$) intensity along the line segment between $\mathbf{p}_j^*$ and $\mathbf{p}_l^*$. Note that this feature is an adapted version of feature V_2 in the supervised learning approach (cf. Eq. 5.5) which was ranked among the best features by the random forest (cf. Table 5.5).

6. Estimate the two class conditional densities $\widehat{P}(V|\mathit{inner})$ and $\widehat{P}(V|\mathit{cross})$ assuming a Gaussian distribution. This is simply done by estimating the mean and variance of V for each of the two Gaussian distributions.

So far, a profile classifier was trained using the input image. The *posterior* probability $\widehat{P}(\mathit{cross}|V)$ assuming equal priors $\widehat{P}(\mathit{cross}) = \widehat{P}(\mathit{inner})$, i. e.:

$$\begin{aligned}\widehat{P}(\mathit{cross}|V) &= \frac{\widehat{P}(\mathit{cross})\widehat{P}(V|\mathit{cross})}{\widehat{P}(\mathit{cross})\widehat{P}(V|\mathit{cross}) + \widehat{P}(\mathit{inner})\widehat{P}(V|\mathit{inner})} \\ &= \frac{\widehat{P}(V|\mathit{cross})}{\widehat{P}(V|\mathit{cross}) + \widehat{P}(V|\mathit{inner})},\end{aligned}$$

is then used to rank each two nearby keypoints (cf. Figure 7.1 (c)). This ranking expresses the probability that they belong to two different cells. In order to reduce runtime, only the three nearest neighbors of each keypoint are considered. The resulting ranks are then used as input for an agglomerative hierarchical clustering with average linkage similar to Chapter 5. The resulting clusters at a cut-off equal to 0.5 (cf. Figure 7.1 (d)) represent the detected cells. Inside each cluster, the arithmetic average of the keypoint coordinates identifies the center of a detected cell.

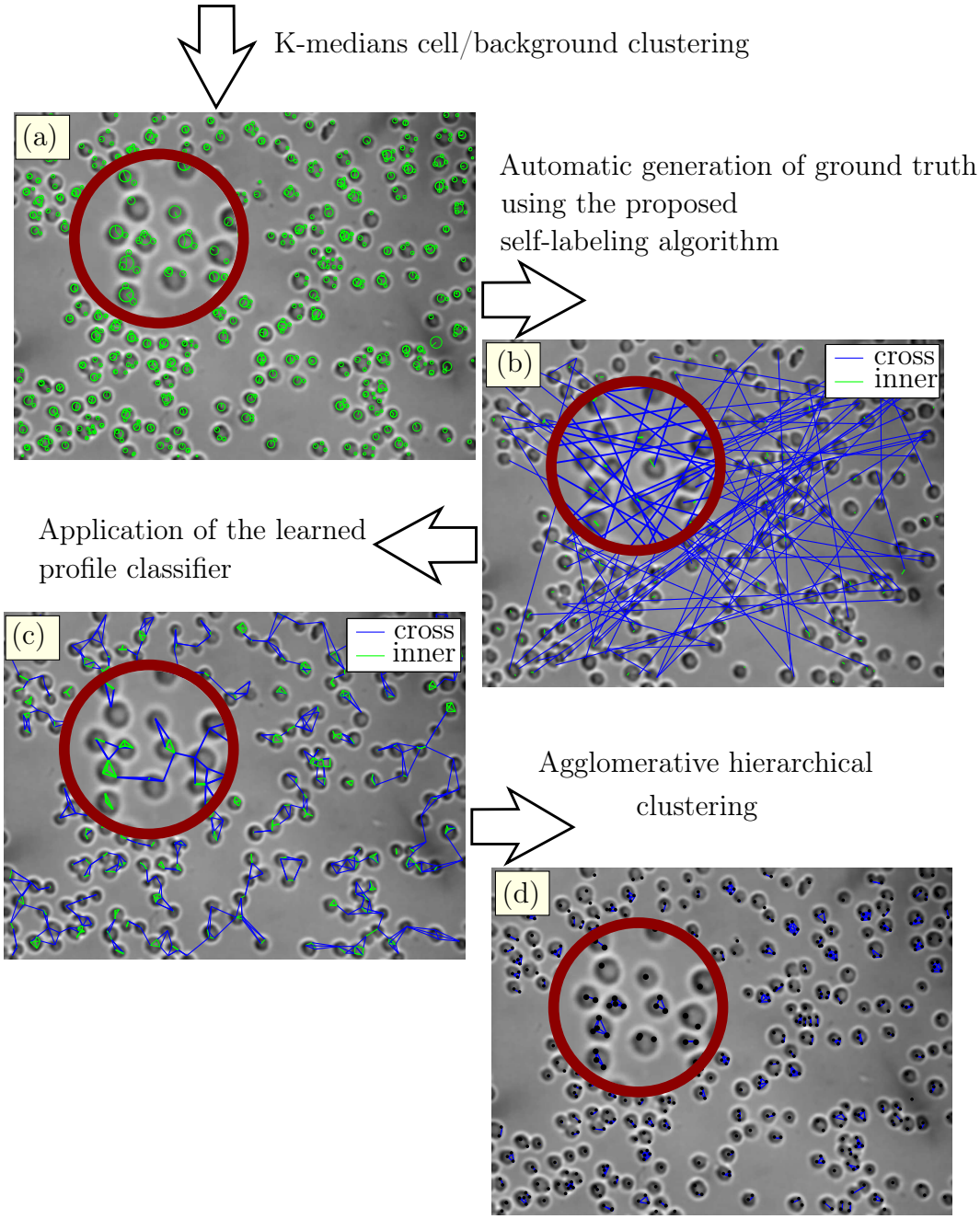


Figure 7.1: Illustration of the cell/cell keypoint clustering. The circle inside each figure shows a magnified view. a) Cell keypoints resulting from the cell/background $\mathbb{k}$-medians clustering. b) Point pairs chosen by the self-labeling algorithm for training a profile classifier. Each pair is indicated by a line segment. c) The learned profile classifier is employed to rank nearby keypoint pairs. The output is probabilistic, but only the binary classification result is shown. d) Result of hierarchical clustering using the ranks obtained from the previous step. Each cluster represents a detected cell.

7.3 Evaluation

As mentioned in Chapter 4, the ground-truth type of all datasets except [Arte 12] is cell border delineation, while in the dataset of [Arte 12] a dot is marked at the center of each cell. This difference in ground-truth representation leads to a difference in the evaluation procedure. In all datasets except [Arte 12], a cell is considered detected if the hit point belongs to the cell mask and the *centeredness error* (cf. Eq. 5.19) is used to assess the deviation from the cell center. This is the evaluation procedure used with the supervised approach in Chapter 5 and Chapter 6. In the dataset of [Arte 12], cell masks are not available. Therefore, a cell is considered detected if the distance to the ground-truth cell center is less than the minimum cell radius. The latter was set after [Arte 12] to 5 pixels. Figure 7.2 exemplifies detection results of our unsupervised approach for the real cell lines of the standard bright field dataset while Figure 7.3 exemplifies the phase contrast results. Quantitative evaluation is given in the next paragraph. The evaluation results of [Pan 10] and [Arte 12] are given according to their corresponding papers.

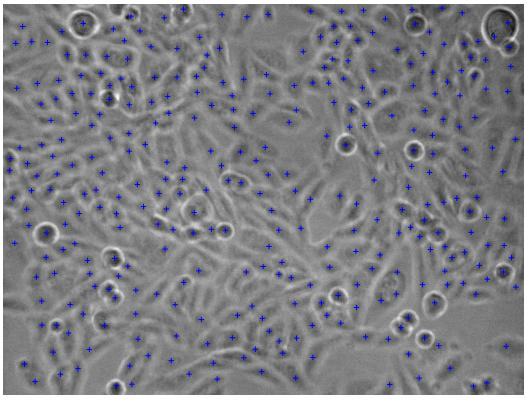
A comparison with our supervised approach presented in Chapter 5 on bright field microscopy is shown in Table 7.1. The figures of the supervised approach in Table 7.1 were obtained by image-wise cross-validation in each cell line: one image per cell line is used for training and the other images of the same cell line are used for testing. The results of the unsupervised approach were obtained by averaging each of the F-measure, time, and centeredness error over images per cell line. A comparison with [Arte 12] and [Pan 10] on phase contrast microscopy is shown in Table 7.2. The shown results of the approaches [Arte 12] and [Pan 10] in Table 7.2 were generated by the hold-out method: [Arte 12] was trained using 11 images and tested on other 11 images. Similarly, [Pan 10] was trained using 10 images and tested on other 10 images. We evaluated our unsupervised approach on the same images which were used for *testing* each of them (the images described in Table 4.2). Tables 7.1 and 7.2 show that the proposed approach is very close in terms of F-measure and centeredness error (when available) to the supervised approaches. However, the proposed unsupervised

		Supervised approach of Chapter 5	Unsupervised proposed approach
F-measure (%)	CHO	84.2	85.1
	L929	86.5	88.3
	Sf21	97.0	89.5
Time (seconds)	CHO	45.9	10.5
	L929	36.7	10.9
	Sf21	40.7	14.4
Centeredness error	CHO	0.48	0.40
	L929	0.38	0.42
	Sf21	0.16	0.23

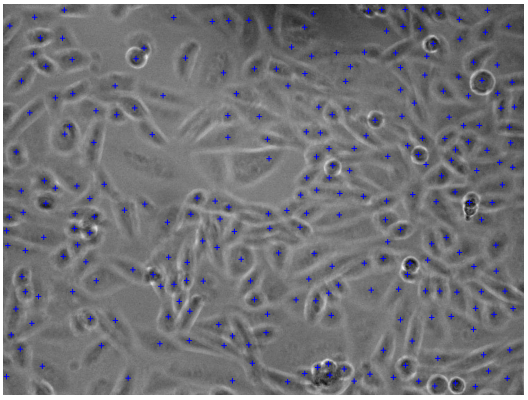
Table 7.1: Comparison with the state-of-the-art on bright field microscopy

		Supervised Pan et al. [Pan 10]	Supervised Arteta et al. [Arte 12]	Unsupervised proposed approach
F-measure (%)	Hela	-	88.0	88.7
	Bovine	94.6	-	86.0
Time (seconds)	Hela	-	30.0	1.5
	Bovine	900.0	-	3.5
Centeredness error	Hela	-	-	-
	Bovine	-	-	0.11

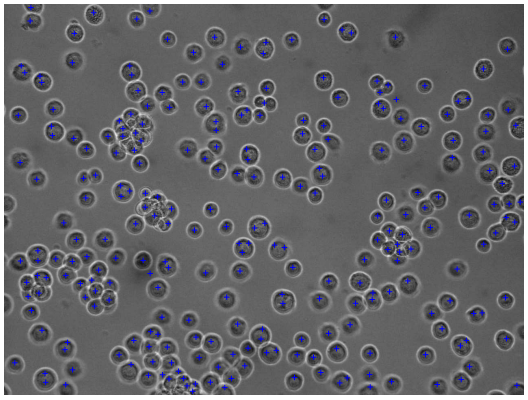
Table 7.2: Comparison with the state-of-the-art on phase contrast microscopy



(a) CHO

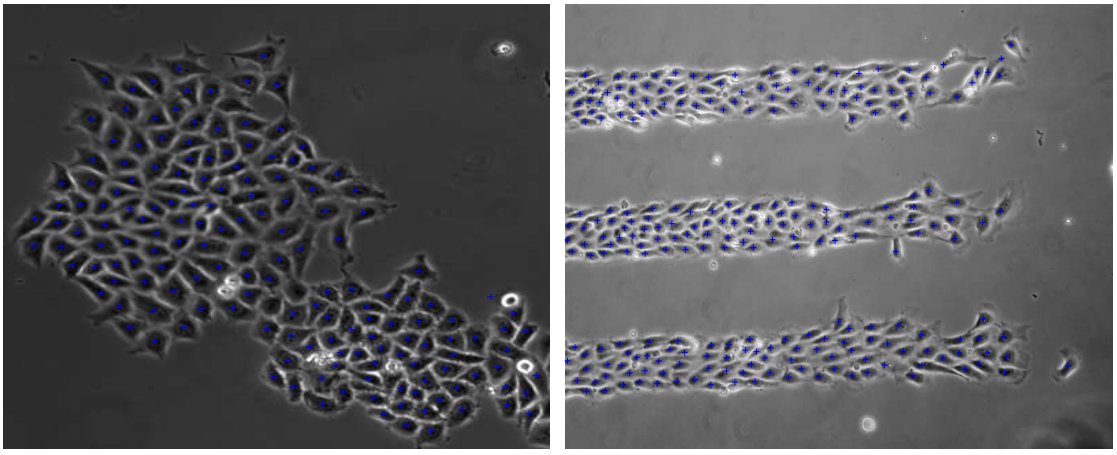


(b) L929



(c) Sf21

Figure 7.2: Samples of the detection results on bright field microscopy: each plus sign marks a detected cell.



(a) Hela (b) Bovine

Figure 7.3: Samples of the detection results on phase contrast microscopy: each plus sign marks a detected cell. The two shown images have different resolutions but they were scaled for display.

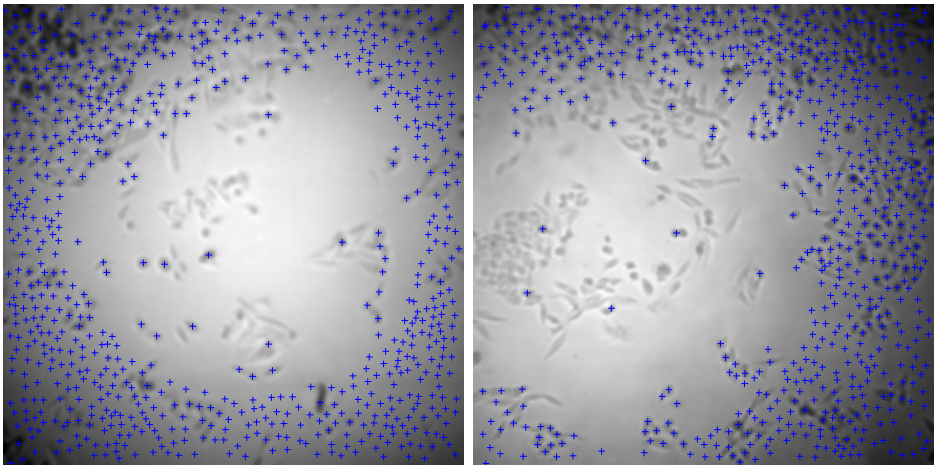
method is much faster especially when compared with the phase contrast approaches where it is one or two orders of magnitude faster.

The blob type was correctly picked for all images by Eq. 7.1. As can be seen in Eq. 7.1, this blob type is decided by the sign function. Therefore, the reliability of the decision is proportional to the absolute value of the sign operand. We observed a little improvement (data not shown) of this reliability when both PCR and scale are used for weighting (as in Eq. 7.2) compared to the case when only the scale is used (as in the supervised approach).

Lastly, we conducted a qualitative evaluation of the unsupervised approach on COSIR images similar to Section 5.8.6. As shown in Figure 7.4a and Figure 7.4b, the unsupervised approach completely fails. This is expected as intensity, which is used as a feature in the $\mathbb{k}$-medians clustering, is not discriminative in the presence of illumination artifacts. In Figure 7.4c and Figure 7.4d, the phase contrast $\mathbb{k}$-medians feature space was used yielding better, but still unsatisfying, results.

7.4 Discussion

Both blob type detection and scale adaptive smoothing were proposed in the supervised approach in Chapter 5. In contrast to Chapter 5, where only keypoints which belong to cells (known from ground truth) are considered, the blob type $\mathrm{BT}_{\mathrm{unsup}}$ was computed in this chapter in unsupervised manner by considering all keypoints. In addition, we use both scale and PCR to weight the keypoint contribution to $\mathrm{BT}_{\mathrm{unsup}}$ whereas only scale is used in the supervised approach. For the scale adaptive smoothing, we use a weighted average instead of the simple arithmetic average used in Chapter 5. In general, we can conclude that SIFT can be successfully employed for *unsupervised* structure-of-interest measurements such as mean scale and dominant curvature direction.



(a) (b)

(c) (d)

Figure 7.4: Evaluation on COSIR images: the unsupervised approach fails on COSIR images, mainly, due to illumination artifacts and insufficient contrast. For the $\mathbb{k}$-medians clustering, the bright field features were used in the upper row and the phase contrast features were used in the lower row (cf. Section 7.2.5). Compared to bright field features, phase contrast features yield better, but still unsatisfying, results.

In the cell/cell clustering step, a self-labeling algorithm was employed to train a ranking classifier. This classifier learns from extreme cases and applies the learned model on intermediate ones. In other words, training and testing feature vectors are drawn from different distributions. Therefore, the features should be chosen carefully so that they do not overfit the training samples. With this in mind, we confined ourselves to use a one-dimensional feature space and a simple generative model. We think that the cell/cell clustering step presented in this chapter may be improved by applying transductive transfer learning techniques. On the other hand, for the possibly less-reliable cell/background clustering, we think that applying transductive learning methods may alleviate the limitations of $\mathbb{k}$-medians. In the self-labeling algorithm, due to the use of SIFT, it was possible to define a scale-invariant notion of the *extreme* cases. Consequently, the algorithm could successfully detect cells in images of different resolutions and/or cell types without any change in the parameter values.

On images obtained by phase contrast microscopy and standard bright field microscopy, the proposed approach achieves detection accuracy which is close to three state-of-the-art supervised cell detection approaches in much less time, without training data, and without manual parameter-tuning. On the other hand, it fails on COSIR images which exhibit less contrast and suffer from illumination artifacts. We thus believe that under standard conditions, the cell detection problem is, to a large extent, solvable by self-supervised techniques which learn from the input image itself even though more research is required in this direction. Nevertheless, when reliability is a main concern or when cell images deviate from standard conditions, supervised approaches are more appropriate.

Chapter 8

Outlook

In this chapter, limitations and shortcomings of the proposed approaches are more closely considered. Moreover, possible extensions are discussed and proposals for further work are made.

8.1 Automatic Defocus Distance Selection

For solving the TIE or estimating the local phase images, the axial intensity derivative is required. This derivative was estimated by subtracting two images at two different focus levels, for instance, $I(0)$ and $I(\Delta z)$. Choosing a proper defocus distance Δz is important for recovering phase values correctly. Mathematically, in order to estimate $\partial I/\partial z$, Δz must approach zero. However, due to noise in physical world, such small distances yield very poor SNR in the resulting derivative image. In order to improve the SNR, a larger Δz value is required as the contrast $\frac{I(\Delta z)-I(0)}{I(0)}$ is proportional to Δz (cf. Eq. 2.23). In fact, using the forward finite differences to estimate the first derivative of $I(z)$ at $z = 0$ has the implicit assumption that $I(z)$ is linear inside $[0, \Delta z]$. Eq. 2.23 reveals that this assumption of linearity is plausible. However, by increasing Δz, smoothing with the point spread function of the optical system also increases leading to non-linear changes which are not modeled in Eq. 2.23. Therefore, an optimal defocus distance which represents a trade-off between SNR and excessive smoothing is required for a reliable estimation of the axial derivative. As pointed out in literature [Soto 03, Wall 10], this optimal defocus distance depends on the frequency spectrum of the imaged object and SNR of the image detector.

The aforementioned optimal defocus distance is ideal for estimating physical phase and low-pass monogenic local phase values. However, cell detection poses a different objective function. In other words, a defocus distance which yields the lowest cell detection error is different from the defocus distance which yields the lowest error in estimating phase values, even though both of them are trade-offs between the same players (SNR and blurring). Moreover, an optimal defocus distance for cell segmentation is, in principle, different from the two previous optimal distances.

In a study of focus curves of phase objects conducted also in context of the COSIR project [Scho 14], it was shown that focus curves show untypical behavior when computed for phase objects. For homogeneous cell cultures, i. e. for cultures in which most cells tend to lie in the same z-plane, both *normalized variance* and *Tenenbaum*

gradient focus measures exhibit a configuration termed *phase effect characteristic* (PEC). This configuration is defined by a minimum at the optical focus position and two maxima, one in each defocusing direction. It was suggested in [Scho 14] to use the images at the two PEC's maxima of the normalized variance for cell detection and the images at the two PEC's maxima of the Tenenbaum gradient for cell segmentation. This was, however, only anticipated and not shown to be working in an experimental evaluation. More research is thus required in this direction.

8.2 Learning to Detect Concave Cells

In order to classify keypoint-pairs as either inner or cross, for each keypoint-pair $(\mathbf{p}_1, \mathbf{p}_2)$, we extract an intensity profile along the line segment between $\mathbf{p}_1$ and $\mathbf{p}_2$. While this is very efficient and easy to implement, going only along the line segment from $\mathbf{p}_1$ to $\mathbf{p}_2$ implicitly assumes that the cell shape is not concave between these two keypoints. To a large extent, this assumption holds for all cell lines used in this thesis. However, in order to make our methods more general, paths between keypoints need to be selected more carefully. One solution for that is to follow the easiest-to-climb path [Pan 09, Pan 10] between the two cell keypoints. This path is found using dynamic programming in the neighborhood of the two corresponding keypoints. The resulting path is thus a local optimum. We suggest investigating the random walker with restarts (RWR) [Kim 08] for computing a dissimilarity measure (probability of belonging to different cells) between nearby keypoints. Since the random walker can follow any path between the two keypoints, the resulting solution is global. In addition, these random paths do not need to be simulated or manually extracted as this problem is known to have an elegant analytic solution.

8.3 Replacing SIFT

SIFT keypoints were heavily used in our methods. While SIFT attains a very good reputation in the computer vision research community, most companies will not prefer incorporating it in their products as it is a patented technique. SIFT is composed of a detector and a descriptor. For our algorithms, we think that it is fairly easy to replace the SIFT descriptor as we already have a rich set of features for keypoint classification. For replacing the SIFT detector, we need a local feature detector which fulfills the following requirements:

1. It detects blobs in intensity. This excludes feature detectors which are based on corner detection such as FAST [Rost 05] and BRISK [Leut 11].

2. It provides a measure of keypoint strength which can replace the $\mathrm{DoG}_{\gamma 1}$ in SIFT.

3. It provides a mechanism to differentiate between black-on-white and white-on-black blobs.

4. Both scale and orientation are specified for each keypoint.

The SIFT's PCR was used in our algorithms as a keypoint feature in the supervised approach (cf. Section 5.3.1) and as a weight in the unsupervised approach (cf. Eq. 7.2). However, according to the resulting feature ranking and experimental evaluation, it is not indispensable and the algorithms will thus not loose much performance by omitting it.

8.4 Reliable Unsupervised Keypoint Learning

In the unsupervised approach presented in Chapter 7, it was shown that it is possible to learn a profile classifier from the input image itself based on automatically generated ground truth. However, the experimental evaluation showed that, mostly due to inefficiency of the cell/background clustering step (cf. Section 7.2.5), the unsupervised approach fails on images suffering from illumination artifacts. One may think of investigating transductive learning techniques for alleviating the limitations of $\mathbb{k}$-medians clustering. This adjustment, albeit expectedly helpful, is not sufficient as it may improve the learning model but does not incorporate information characteristic to cell images. Such information was employed, for instance, by the self-labeling algorithm for profile learning (cf. Section 7.2.6). Inspired by the success of this self-labeling algorithm, one may wonder whether it is possible, based on some fair assumptions, to design a learning algorithm in the same spirit for cell/background separation which does not require manually-labeled ground truth.

8.5 Extracting Features From the SIFT GSS

In the proposed supervised approach of Chapter 5, both keypoint features and profile features were extracted from a smoothed image. The standard deviation of the Gaussian kernel used for smoothing was set to mean scale $\bar{\sigma}$ of the one-sided keypoints of the input image, and the resulting operation was thus called scale adaptive smoothing (cf. Section 5.4.3). This has the advantage of implicitly incorporating relevant structure size into the feature extraction pipeline. It was also shown that this step, i. e. extracting the features from the smoothed image $I_{\bar{\sigma}}$, improves the accuracy of cell detection (cf. Section 5.8.2), even though in our image database, the standard deviation of cell size in the same image[1] is not small. For instance, in the L929 cell line, the major axis length of cells in a single image is estimated as 71.74 ± 33.16 pixels[2] and the minor axis length is 32.96 ± 10.42 pixels.

As alternative to this global strategy, one may think of extracting the features from the SIFT GSS. For example, intensity stencil, ray features, and variance map at a keypoint $(x, y, \sigma, \vartheta)$ can be extracted from the level $L(., ., \sigma)$ in the GSS (instead of $I_{\bar{\sigma}}$). Moreover, profile features for an intensity profile between $\mathbf{p}_1$ and $\mathbf{p}_2$ can be

[1] Since $\bar{\sigma}$ is computed for each image, the standard deviation of cell size in the considered image (not in the entire training data) is what matters here.

[2] This is computed as follows: in each L929 image, the mean and standard deviation of major axis length of the cells in this image are computed. The value of 71.74 is the average of the aforementioned means over all L929 images while the value of 33.16 is the average of the standard deviations. Estimating the minor axis length is performed in a similar way.

extracted from a GSS level which is relevant for both keypoints, say $L(.,.,\frac{\sigma_1+\sigma_2}{2})$ (approximated to the closest available level in the GSS). On one side, this strategy is more scale-specific and one may thus expect improvements in detection accuracy. On the other side, there is no clue about the relevant structures in the image. Consider, for instance, two keypoints at a very low scale, the decisions (cell/background and inner/cross) will be made based on image details which are likely to be too small compared to cell size. This results in lack of global information. More investigation is thus required in order to gain the advantages of both strategies.

8.6 Application on Other Microscopic Modalities

In this thesis, we presented cell detection results on images acquired using phase contrast microscopy, standard bright field microscopy, and COSIR hardware. In addition, there are experiments on other microscopic modalities not reported so far. For instance, we applied the supervised approach of Chapter 5 for nuclei detection in fluorescence microscopy and also for molecule detection (nano-scale objects) in scanning tunneling microscopy. Similar to the experiments reported in this thesis, one image was used for training and no parameter tuning was conducted. These experiments provided good qualitative results but they were not performed on a large scale. Therefore, they can be seen as a proof-of-concept and serve as indicators that the supervised pipeline can be employed in diverse tasks where detection of blob-like structures is involved.

8.7 Adaptation of Standard Features Based on Keypoints

In Section 5.3, keypoint-based forms of variance maps, ray features, and intensity stencils were proposed. As mentioned earlier, this extraction scheme is sparse, scale-invariant, orientation-invariant, and enables feature parameters to be tailored in a meaningful way based on a relevant scale and orientation. While this is similar to the concept of keypoint *descriptors* in the literature, we think it can be more widely applied in application-dependent tasks to other standard feature sets, say for instance, Haralick texture features [Hara 73] and Haar-like features [Papa 98, Viol 01].

8.8 Cell Viability Determination

As explained in Section 2.4, experiments of cell viability determination are typically performed in laboratories using staining. A widely-accepted standard for this process is the PI-staining (cf. Figure 2.8). It was shown in literature that supervised machine learning methods can separate viable from non-viable cells based on visual appearance as depicted in *unstained* cell images. In this case, staining is required only for obtaining ground-truth labels to train the involved classifiers. In fact, this automatic viability determination from unstained images has a great practical impact because of at least two reasons: Firstly, measurements of viable cell concentration play a

vital role in several important fields including biology, medicine, and pharmaceutics. Secondly, the manual methods widely used for this purpose are time-consuming, laborious, and not accurate.

There are several published works which tackle the problem of automatic viability classification based on unstained cell images. This includes viability determination applied on: 1) cells from swine tissues in phase contrast microscopy using texture features [Malp 03], 2) yeast cells in dark field microscopy using energy and entropy of wavelet sub-images [Wei 08], 3) A20.2J murine B-cell lymphoma cells in bright field microscopy using pixel patches [Long 06].

In context of the COSIR project, we also investigated cell viability determination on both phase contrast and bright field images of CHO cells [Van 13]. An important consequence of this work is that, at least on the datasets employed for this study, viability can be determined from unstained bright field images with accuracy comparable to that obtained from phase contrast images. This was the case with different feature sets including the keypoint features of Section 5.3.1. For extending this work, we make the following points:

- Both [Van 13] and [Wei 08], even though being done with different cell types and microscopic modalities, share a common observation: dead cells tend to be more homogeneous and show less cellular details. Therefore, incorporating more features which capture this property may improve classification accuracy. To give just one example, in [Wei 08], energy and entropy of wavelet-decomposition sub-images (up to the second level) features were extracted from a fixed-size patch around the center of each detected cell. These features were shown to be discriminative on the dark field images. Adapting them to the keypoint-based feature extraction scheme in our pipeline may thus improve the viability classification rate on bright field images.

- Is it more appropriate to perform cell viability determination at the cell level (after cell detection) or at the keypoint level? The answer depends on how each of them is implemented. The first choice is more intuitive. On the other hand, the second case seems to be convenient in our supervised cell detection pipeline for three reasons: 1) It is straightforward to extend the keypoint classifier for three classes (background, viable, and non-viable) instead of the current two classes (background and cell) as the random forest is inherently a multi-class classifier. 2) The keypoint features of Section 5.3.1 worked pretty well for the viability determination experiments in [Van 13]. 3) At the end of cell detection in our pipeline, one obtains a cluster of keypoints inside each cell. In one cluster, there will be, in principle, both viable and non-viable keypoints. It is then possible to aggregate the results of the same cluster, either to obtain a confidence index or to make a more reliable decision. The values of keypoint scale, orientation, $\mathrm{DoG}_{\gamma 1}$, and PCR can be employed to weight individual keypoint contributions to the final decision.

- As in any study, soundness and generalizability of the derived statements are dependent on diversity of the considered data. For viability determination, this seems to be of special importance as cell death cannot be uniquely and

unambiguously defined and it may manifest itself in different morphological behaviors. For instance, while the lack of cellular details was a prominent feature of non-viable cells in [Wei 08] and [Van 13], the fragmentation into small subcellular structures was the the main observation made on dead cells in [Malp 03].

8.9 Simulation of Cell Image Stacks in Bright Field Microscopy

In this thesis, the software package SIMCEP [Lehm 07] was employed to simulate additional cell images along with their ground-truth masks. This software was originally developed for fluorescence microscopy. As mentioned in Section 4.1.3, it was used in our experiments due to the unavailability of a proper alternative for bright field microscopy. We suggest to extend SIMCEP for simulating cell image stacks in bright field microscopy as follows: it can be assumed that a simulated SIMCEP image without cytoplasm (cf. Figure 4.1d) represents a phase map $\phi_{\text{sim}}(x, y)$ of a very thin sample. In this case, the image stack can be theoretically obtained by solving the simplified form of the transport of intensity equation (cf. Eq. 2.23) for $I(z)$:

$$I_{\text{sim}}^{\text{ideal}}(x, y, z) = \frac{-I_0}{k} z \nabla_{\perp}^2 \phi_{\text{sim}}(x, y) + I_0, \tag{8.1}$$

where $I_0 = I_{\text{sim}}^{\text{ideal}}(x, y, 0) = \text{constant}$ and z is the distance to the focus position which is assumed to be at $z = 0$. Eq. 8.1 describes an ideal image because blurring by point spread function of the optical system is not modeled. Incorporating a point spread function PSF_{sim} in the simulation yields:

$$\begin{aligned} I_{\text{sim}}(x, y, z) &= I_{\text{sim}}^{\text{ideal}}(x, y, z) * \text{PSF}_{\text{sim}}(x, y, z) \\ &= \left(\frac{-I_0}{k} z \nabla_{\perp}^2 \phi_{\text{sim}}(x, y) + I_0 \right) * \text{PSF}_{\text{sim}}(x, y, z) \cdot \end{aligned} \tag{8.2}$$

For a diffraction-limited optical system, i. e. a system whose resolving power is limited only by the wavelength of light, the PSF_{sim} is given by an Airy pattern (cf. Section 2.7). A more realistic PSF_{sim} would also model the aberrations caused by hardware imperfections and the deviations resulting from experimental setup in laboratory. From this perspective, the PSF_{sim} model of Gibson & Lanni [Gibs 92] seems to be a good candidate, especially that it can be used in a bright field setup [Tadr 10]. For the purpose of this discussion, it is sufficient to assume a Gaussian approximation. In this case, the standard deviation of the Gaussian kernel σ_{psf} is linearly increasing with the defocus distance [Ague 08]:

$$\sigma_{\text{psf}}(z) = C_1 \, |z| + C_2, \tag{8.3}$$

where C_1 and C_2 are related to microscope parameters and experimental setup. They can be thus considered as part of the simulation parameters. By using this Gaussian

approximation model for $\mathrm{PSF}_{\mathrm{sim}}$ in Eq. 8.2, the simulated bright field stack can be given as:

$$\begin{aligned} I_{\mathrm{sim}}(x,y,z) &= \left(\frac{-I_0}{k} z \nabla_{\perp}^2 \phi_{\mathrm{sim}}(x,y) + I_0\right) * s(x,y,\sigma_{\mathrm{psf}}^2) \\ &= \left(\frac{-I_0}{k} z \nabla_{\perp}^2 \phi_{\mathrm{sim}}(x,y) + I_0\right) * \frac{1}{2\pi(C_1\,|z| + C_2)^2} e^{-\frac{x^2+y^2}{2(C_1|z|+C_2)^2}}, \end{aligned} \tag{8.4}$$

where s is the Gaussian kernel. Note that k, I_0, C_1, and C_2 are constants which can be set by the user along with the other simulation parameters.

If the simulated phase map $\phi_{\mathrm{sim}}(x,y)$ contains noise in background, $\nabla_{\perp}^2 \phi_{\mathrm{sim}}(x,y)$ in the background is not zero. In this situation, inside the same lateral plane (same z value), the simulated increased contrast by defocusing is nothing more than a scaling by a constant z followed by smoothing, which is, unlike natural defocusing, does not improve the contrast. Therefore, this scaling in the simulation has to be done only for cell pixels as to mimic the natural defocusing process correctly.

Chapter 9

Summary

In this thesis, automatic unstained cell detection on bright field microscope images was investigated. The research was done in context of the interdisciplinary research project COSIR. COSIR aimed at developing a new microscopic hardware having the following attractive feature: it can be placed inside the incubator where cells can be cultivated under ideal physical conditions. The device is composed of 24 channels, each of which delivers an image of a single well from a 24-wells microtiter-plate. A bright field microscopic pipeline was implemented in each channel. Due to size limitations and other manufacturing challenges, the bright field technique was preferred over other microscopic modalities. In addition to introducing the COSIR project in Chapter 1, a review of the state-of-the-art in unstained cell detection was presented and the contributions of this work to the progress of research were stated.

In Chapter 2, the most widely-used microscopic image modalities were presented. The main focus was on light microscopy, i. e. microscopic techniques which employ visible light in order to form an image of a given specimen. Magnification mechanisms in these light microscopy techniques are conceptually similar and can be explained using the thin lens model and the compound microscope principle. On the other hand, the interpretation of information in the acquired images is modality-dependent. In bright field microscopy, image formation is based on the attenuation of light amplitude due to light absorbing induced by the imaged specimen. A fluorescence microscope, on the other hand, depicts the fluorescent radiation of fluorescent materials injected in the specimen. Furthermore, in phase contrast microscopy, the obtained contrast is a result of variations in light phase upon passing through different parts of the specimen, which are assumed to vary in refractive index and/or thickness. These phase variations cannot be detected by a CCD chip or human eye, and hence, in order to form an image, phase variations are converted to amplitude changes using the so-called Zernike's trick. Related to both phase contrast and bright field techniques, the so-called quantitative phase microscopy is increasingly gaining more interest. In short, it yields phase information computationally from bright field images obviating the need to the expensive and complicated phase contrast hardware. An additional advantage of this technique is that the obtained phase information is *quantitative*. This enables reconstruction of refractive index or thickness profile from the computed phase maps. The transport of intensity equation, being a promising quantitative

phase technique, was explained in Chapter 2 and put in context with the more general Helmholtz's equation and wave equation.

In Chapter 3, theoretical concepts concerning blob detection, random forests, and agglomerative hierarchical clustering were presented. Our cell detection methods belong to the interest-point based methods. These interest points are detected as scale-space extrema of a blobness measure such as the Laplacian of Gaussian or the difference of Gaussians. Comparing the raw derivative values at different scales (for finding scale-space extrema) is inappropriate as the derivative amplitude always decreases with smoothing (increasing scale t). In the γ-normalized derivatives, however, the decrease of derivative amplitude by smoothing is compensated by multiplying the mth-order derivative with $t^{\gamma m/2}$. Lindeberg showed that a γ-normalized derivative attains an extremum at a scale related to structure size. Moreover, when $\gamma = 1$, the derivative amplitude at the extremum is scale-independent. These principles were employed and extended in SIFT along with carefully-chosen parameter values and intensive evaluations in order to detect interest points reliably and efficiently. In addition to blob detection, in Chapter 3, the concepts of decision trees, bagging, feature random selection, and random forests were explained. Moreover, the basic principles of agglomerative hierarchical clustering were discussed. Of special importance is the Lance-Williams model which expresses a linkage method using a recursive equation so that it can be computed efficiently. In addition, the so-called monotonicity of the resulting clustering trees can be guaranteed if the model coefficients fulfill certain requirements.

In Chapter 4, we described the image materials used for evaluating the proposed cell detection algorithms. We acquired standard bright field images of three cell lines and simulated other two cell lines. These images along with their ground-truth masks are available online so that they can be of benefit for the community. Moreover, COSIR images were also employed in a qualitative evaluation. The COSIR image set was obtained at the end of the financial support period of the COSIR project, as to reflect the most recent status of the hardware. In addition to the bright field images acquired by us, phase contrast images collected by other research groups were considered for the evaluation.

In Chapter 5, a new supervised cell detection approach is introduced. We employed a point-based method where a two-class problem is solved first in order to classify keypoints as belonging to background or cells. Cell keypoints are then grouped together by another classifier so that it is possible to rank each pair of keypoints in the sense whether they belong to the same cell or to different cells. While this rough pipeline was used in the literature of phase contrast imaging, in this thesis, considerable novel contributions were added and heavy experimental evaluations were conducted (cf. Section 1.3). Standard pixel-based feature sets such as ray features, intensity stencils, and variance maps were adapted so that they can be extracted in a keypoint-based manner. This adaptation makes the aforementioned feature sets sparse, scale-invariant, and rotation-invariant. In addition, their related parameters can be chosen in a meaningful way according to a relevant scale and orientation. The probabilistic result of the profile classifier, indicating the probability that a pair of keypoints belongs to two different cells, can be seen as a dissimilarity measure. This dissimilarity is used as input for an agglomerative hierarchical clustering procedure.

Instead of the standard linkage methods, we employ a novel SIFT-based method which incorporates the information obtained by SIFT keypoints in the linkage. We proved that this linkage method is combinatorial, i. e. it can be written as a Lance-Williams recursive equation, and hence, the clustering procedure can be performed efficiently. Moreover, we showed that it is monotonic, and the resulting dendrograms are thus interpretable.

A special care was given to robustness with respect to illumination artifacts. This robustness was achieved by: 1) Making keypoint features and profile features invariant to local shift of intensity. 2) Refraining from taking decisions based on large image regions, as in this case the illumination field cannot be approximated by a constant. For instance, only short-length intensity profiles are extracted and the customized linkage method gives a weight to each intensity profile inversely proportional to the squared length of the line segment between the corresponding keypoints (cf. Eq. 5.8). Learning to extract short-length profiles was done in a scale-invariant manner so that the maximum inner profile length (cf. Section 5.4.2) can be learned from training images having diverse resolutions and applied then on testing images of different resolutions.

In order to test the approach's robustness, it was trained with images at specific scale, orientation, and illumination conditions. The trained model was then tested on images at completely different scales, orientations, and illumination artifacts. The results show that the proposed approach is very robust with respect to these factors. A general additional consequence from the evaluation is that the algorithm can learn to detect cells efficiently from a relatively small training dataset: in our experiments, one image per cell line was used for training, while the remaining images were used for testing. Moreover, the system was trained and tested in a completely automatic manner since no manual parameter tuning was required.

We pointed out in the introduction that an interesting link between physical phase of light and local phase was introduced recently in the literature. Since this was not sufficiently studied by other researchers, we investigated part of its applications for bright field image analysis. The aforementioned link can be briefly described as follows: the local phase of the monogenic signal framework can be used to approximate the solution of the transport of intensity equation, which is the physical phase of light. This approximation is valid under two conditions: 1) The axial intensity derivative is used as input of the monogenic framework. 2) A low-pass filter which approximates the inverse Laplacian is utilized instead of the typically-used band-pass filter. In Chapter 6, we explained this link in detail and clarified its plausibility. We showed with several experiments that local phase is more discriminative than defocused images in pixelwise cell/background classification. Moreover, we showed that both TIE-solution images and local phase images can be successfully employed to improve joint learning of suspended and adherent cell detection. When we trained our supervised approach presented in Chapter 5 on both suspended and adherent cells (joint learning), the detection accuracy was reduced compared to the case of separate learning where the algorithm is trained and then tested on each cell type alone. This reduction in detection accuracy is plausible due to the remarkable difference between suspended and adherent cells in contrast and visual appearance. On the other hand, when the keypoint features and the profile features of Chapter 5 were extracted from

a TIE-solution image or from a local phase image instead of a defocused image, the generalization ability of the algorithm was considerably improved yielding a result which is almost as good as the separate learning case. One may question the importance of this generalizability as follows: it does not hurt to train the supervised approach of Chapter 5 on adherent cells alone and then applying the trained system on test images which contain only adherent cells. Likewise, this can be done with suspended cells. However, adherent and suspended cells coexist in cell cultures even though this is not very well reflected in our image materials. In addition, the states of adherence and suspension are two terminal cases with shades of gray between them. This generalizability is thus relevant in practice.

In Chapter 7, we investigated unsupervised cell detection. This was seen in the following perspective: if images do not exhibit severe artifacts and reliability can be compromised for having a labeling-free cell detection system, then unsupervised learning is easier to apply and manage. In this approach, the dominant curvature type, i. e. whether cells are dominated by black-on-white or white-on-black blobs, was successfully determined without supervision. Keypoints were categorized into cell and background clusters using $\mathbb{k}$-medians clustering. Afterwards, in order to separate intensity profiles into *inner* and *cross* similar to the supervised approach, a profile classifier was employed. However, this classifier was trained without manually-labeled ground truth. A new self-labeling algorithm was suggested which can employ SIFT keypoints and the cells available in the input image to generate ground truth. This was motivated by the following intuition: we can assume ground truth (cross or inner) for the very *long* line segments and also for the very *short* ones. A very long line segment between two cell keypoints will cross the borders of at least one cell and it can be thus labeled as cross. On the other hand, a very short line segment, or more specifically, a line segment inside a single cell keypoint, can be labeled as inner. This is justified by the observation that a cell keypoint, being an indicator of a structure inside a cell, is very unlikely to span more than one cell. A profile classifier is trained based on these extreme cases and applied then on the intermediate ones. Due to the use of SIFT, this notion of long and short line segments can be defined in a scale-invariant way which is, to a large extent, independent from image resolution and cell type. The unsupervised approach was applied successfully on phase contrast and standard bright field images and was comparable, with a bit inferior accuracy but much shorter detection time, to state-of-the-art supervised approaches including the one suggested in Chapter 5. However, it failed on COSIR images which exhibit lower contrast and suffer from dominant illumination artifacts while our supervised approach was able to detect cells reliably and efficiently on those images.

Chapter 8 contains proposals for further work. Suggestions were made to extend the supervised approach using random walks, SIFT GSS, and cell viability determination. Moreover, discussions and recommendations were given regarding replacing the patented SIFT technique, automatic selection of defocus distance, reliable unsupervised cell/background separation, and simulation of bright field stacks.

List of Figures

List of Tables

Bibliography

[Ager 03] U. Agero, C. H. Monken, C. Ropert, R. T. Gazzinelli, and O. N. Mesquita. "Cell surface fluctuations studied with defocusing microscopy". *Physical Review E*, Vol. 67, No. 5, p. 051904, May 2003.

[Ague 08] F. Aguet, D. Van De Ville, and M. Unser. "Model-based 2.5-D deconvolution for extended depth of field in brightfield microscopy". *Image Processing, IEEE Transactions on*, Vol. 17, No. 7, pp. 1144–1153, July 2008.

[Albe 05] B. Alberts, D. Bray, K. Hopkin, A. Johnson, J. Lewis, M. Raff, K. Roberts, and P. Walter. *Lehrbuch der Molekularen Zellbiologie*. Wiley-VCH, Weinheim, Germany, 2005.

[Ali 07] R. Ali, M. Gooding, M. Christlieb, and M. Brady. "Phase-based segmentation of cells from brightfield microscopy". In: *Proceedings of the IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pp. 57–60, Arlington, VA, USA, April 2007.

[Ali 10] R. Ali, T. Szilagyi, M. Gooding, M. Christlieb, and M. Brady. "On the use of low-pass filters for image processing with inverse Laplacian models". *Journal of Mathematical Imaging and Vision*, Vol. 43, No. 2, pp. 1–10, June 2010.

[Ali 12] R. Ali, M. Gooding, T. Szilágyi, B. Vojnovic, M. Christlieb, and M. Brady. "Automatic segmentation of adherent biological cell boundaries and nuclei from brightfield microscopy images". *Machine Vision and Applications*, Vol. 23, No. 4, pp. 607–621, July 2012.

[Arnd 89] J. Arndt-Jovin and T. M. Jovin. "Fluorescence labeling and microscopy of DNA". *Methods in Cell Biology*, Vol. 30, pp. 417–448, 1989.

[Arte 12] C. Arteta, V. Lempitsky, J. Noble, and A. Zisserman. "Learning to detect cells using non-overlapping extremal regions". In: N. Ayache, H. Delingette, P. Golland, and K. Mori, Eds., *Medical Image Computing and Computer-Assisted Intervention MICCAI 2012*, pp. 348–356, Springer Berlin Heidelberg, Nice, France, October 2012.

[Bata 81] V. Batagelj. "Note on ultrametric hierarchical clustering algorithms". *Psychometrika*, Vol. 46, No. 3, pp. 351–352, September 1981.

[Bay 08] H. Bay, A. Ess, T. Tuytelaars, and L. Van Gool. "Speeded-up robust features (SURF)". *Computer Vision and Image Understanding*, Vol. 110, No. 3, pp. 346–359, June 2008.

[Beca 11] G. Becattini, L. Mattos, and D. Caldwell. "A novel framework for automated targeting of unstained living cells in bright field microscopy". In:

S. Wright and X. Pan and M. Liebling, Ed., *Proceedings of the IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, pp. 195–198, Chicago, IL, USA, April 2011.

[Bier 15] B. Bier, F. Mualla, S. Steidl, C. Bohr, H. Neumann, A. Maier, and J. Hornegger. "Band-pass filter design by segmentation in frequency domain for detection of epithelial cells in endomicroscope images". In: Heinz Handels and Thomas M. Deserno and Hans-Peter Meinzer and Thomas Tolxdorff, Ed., *Bildverarbeitung für die Medizin 2015 - Algorithmen, Systeme, Anwendungen*, pp. 413–418, Lübeck, Germany, March 2015.

[Bouk 04] D. Boukerroui, J. A. Noble, and M. Brady. "On the choice of band-pass quadrature filters". *Journal of Mathematical Imaging and Vision*, Vol. 21, No. 1, pp. 53–80, July 2004.

[Brei 01] L. Breiman. "Random Forests". *Machine Learning*, Vol. 45, No. 1, pp. 5–32, October 2001.

[Brei 02] L. Breiman. "Manual on setting up, using, and understanding random forests v3. 1". Tech. Rep., Statistics Department University of California Berkeley, CA, USA, 2002.

[Brei 84] L. Breiman, J. Friedman, R. Olshen, and C. Stone. *Classification and Regression Trees*. Wadsworth and Brooks, Monterey, CA, 1984.

[Brei 96] L. Breiman. "Bagging predictors". *Machine learning*, Vol. 24, No. 2, pp. 123–140, August 1996.

[Cann 86] J. Canny. "A computational approach to edge detection". *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 8, No. 6, pp. 679–698, November 1986.

[Chan 09] D. E. Chandler and R. W. Roberson. *Bioimaging: Current Concepts in Light and Electron Microscopy*. Jones & Bartlett Publishers, Bolingbrook, IL, USA, 2009.

[Chan 11] C.-C. Chang and C.-J. Lin. "LIBSVM: A library for support vector machines". *ACM Transactions on Intelligent Systems and Technology*, Vol. 2, No. 3, pp. 1–27, April 2011. Software available at `http://www.csie.ntu.edu.tw/~cjlin/libsvm`.

[Chen 04] C. Chen, A. Liaw, and L. Breiman. "Using random forest to learn imbalanced data". Tech. Rep. 666, Statistics Department University of California Berkeley, CA, USA, July 2004.

[Cox 12] G. Cox. *Optical Imaging Techniques in Cell Biology*. CRC Press, Boca Raton, FL, USA, 2012.

[Curl 04] C. Curl, T. Harris, P. Harris, B. Allman, C. Bellair, A. Stewart, and L. Delbridge. "Quantitative phase microscopy: a new tool for measurement of cell culture growth and confluency in situ". *Pflügers Archiv European Journal of Physiology*, Vol. 448, No. 4, pp. 462–468, July 2004.

[De B 65] L. De Broglie. "The wave nature of the electron (Nobel lecture, December, 1929)". *Nobel lectures, Physics 1922-1941*, pp. 244–256, 1965.

[DiMa 11] C. A. DiMarzio. *Optics for Engineers*. CRC Press, Boca Raton, FL, USA, 2011.

[Domi 00] P. Domingos and G. Hulten. "Mining high-speed data streams". In: *Proceedings of the Sixth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 71–80, ACM, New York, NY, USA, August 2000.

[Eger 05] R. F. Egerton. *Physical Principles of Electron Microscopy.* Springer, New York, NY, USA, 2005.

[Fels 01] M. Felsberg and G. Sommer. "The monogenic signal". *IEEE Transactions on Signal Processing*, Vol. 49, No. 12, pp. 3136–3144, December 2001.

[Feyn 63] R. Feynman, R. Leighton, and M. Sands. *The Feynman Lectures on Physics.* Vol. 1, Addison-Wesley, Boston, USA, second Ed., 1963.

[Fora 02] M. Foracchia and A. Ruggeri. "Estimating cell density in corneal endothelium by means of Fourier analysis". In: *Engineering in Medicine and Biology, 2002. 24th Annual Conference and the Annual Fall Meeting of the Biomedical Engineering Society, EMBS/BMES Conference. Proceedings of the Second Joint*, pp. 1097–1098, IEEE, Houston, TX, USA, October 2002.

[Freu 97] Y. Freund and R. E. Schapire. "A decision-theoretic generalization of online learning and an application to boosting". *Journal of Computer and System Sciences*, Vol. 55, No. 1, pp. 119–139, August 1997.

[Galb 11] C. G. Galbraith and J. A. Galbraith. "Super-resolution microscopy at a glance". *Journal of Cell Science*, Vol. 124, No. 10, pp. 1607–1611, May 2011.

[Gibs 92] S. F. Gibson and F. Lanni. "Experimental test of an analytical model of aberration in an oil-immersion objective lens used in three-dimensional light microscopy". *J. Opt. Soc. Am. A*, Vol. 9, No. 1, pp. 154–166, January 1992.

[Gil 03] D. Gil, P. Radeva, and F. Vilarino. "Anisotropic contour completion". In: *Proceedings of the International Conference on Image Processing*, pp. 869–872, Barcelona, Spain, September 2003.

[Glck 09] J. Glückstad and D. Palima. *Generalized phase contrast: Applications in optics and photonics. Springer Series in Optical Sciences*, Springer Netherlands, 2009.

[Good 96] J. Goodman. *Introduction to Fourier Optics.* McGraw-Hill, New York, NY, USA, second Ed., 1996.

[Hami 44] W. R. Hamilton. "On quaternions, or on a new system of imaginaries in algebra". *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, Vol. 25, No. 163, pp. 10–13, 1844.

[Hara 73] R. Haralick, K. Shanmugam, and I. Dinstein. "Textural features for image classification". *Systems, Man and Cybernetics, IEEE Transactions on*, Vol. SMC-3, No. 6, pp. 610–621, November 1973.

[Hast 09] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning.* Springer, New York, NY, USA, 2009.

[Heis 10] B. Heise. *Phase-based imaging and image processing in optical coherence tomography applications.* PhD thesis, Johannes Kepler Universität Linz, 2010.

[Hell 09] S. W. Hell. "Microscopy and its focal switch". *Nature methods*, Vol. 6, No. 1, pp. 24–32, January 2009.

[Jian 10] R. Jiang, D. Crookes, N. Luo, and M. Davidson. "Live cell tracking using SIFT features in DIC microscopic videos". *IEEE Transactions on Biomedical Engineering*, Vol. 57, No. 9, pp. 2219–2228, September 2010.

[Jin 03] R. Jin, Y. Liu, L. Si, J. Carbonell, and A. G. Hauptmann. "A new boosting algorithm using input-dependent regularizer". In: *Proceedings of Twentieth International Conference on Machine Learning (ICML) 03*, AAAI Press, Washington, DC, USA, August 2003.

[Jurr 10] E. Jurrus, A. Paiva, S. Watanabe, J. Anderson, B. Jones, R. Whitaker, E. Jorgensen, R. Marc, and T. Tasdizen. "Detection of neuron membranes in electron microscopy images using a serial neural network architecture". *Medical Image Analysis*, Vol. 14, No. 6, pp. 770–783, December 2010.

[Khal 11] M. Khalilia, S. Chakraborty, and M. Popescu. "Predicting disease risks from highly imbalanced data using random forest". *BMC Medical Informatics and Decision Making*, Vol. 11, No. 1, pp. 1–13, December 2011.

[Khos 07] T. Khoshgoftaar, M. Golawala, and J. Van Hulse. "An empirical study of learning from imbalanced data using random forest". In: *Proceedings of the IEEE International Conference on Tools with Artificial Intelligence*, pp. 310–317, Patras, Greece, October 2007.

[Kim 08] T. H. Kim, K. M. Lee, and S. U. Lee. "Generative image segmentation using random walks with restart". In: *10th European Conference on Computer Vision (ECCV)*, pp. 264–275, Springer, Marseille, France, October 2008.

[Koth 05] U. Köthe and M. Felsberg. "Riesz-transforms versus derivatives: on the relationship between the boundary tensor and the energy tensor". In: R. Kimmel, N. Sochen, and J. Weickert, Eds., *Scale Space and PDE Methods in Computer Vision*, pp. 179–191, Springer, Hofgeismar, Germany, April 2005.

[Kwek 02] S. Kwek and C. Nguyen. "iBoost: Boosting using an instance-based exponential weighting scheme". In: T. Elomaa, H. Mannila, and H. Toivonen, Eds., *13th European Conference on Machine Learning (ECML)*, pp. 245–257, Springer Berlin Heidelberg, Helsinki, Finland, August 2002.

[Lace 99] A. Lacey. *Light Microscopy in Biology: A Practical Approach*. Oxford University Press, Oxford, UK, second Ed., 1999.

[Lanc 66] G. N. Lance and W. T. Williams. "A generalized sorting strategy for computer classifications". *Nature*, Vol. 212, No. 5058, p. 218, October 1966.

[Lanc 67] G. N. Lance and W. T. Williams. "A general theory of classificatory sorting strategies: 1. hierarchical systems". *The Computer Journal*, Vol. 9, No. 4, pp. 373–380, February 1967.

[Lehm 07] A. Lehmussola, P. Ruusuvuori, J. Selinummi, H. Huttunen, and O. Yli-Harja. "Computational framework for simulating fluorescence microscope images with cell populations". *IEEE Transactions on Medical Imaging*, Vol. 26, No. 7, pp. 1010–1016, July 2007.

[Leut 11] S. Leutenegger, M. Chli, and R. Y. Siegwart. "BRISK: Binary robust invariant scalable keypoints". In: *Computer Vision (ICCV), 2011 IEEE International Conference on*, pp. 2548–2555, IEEE, Barcelona, Spain, November 2011.

[Li 08] K. Li, M. Chen, T. Kanade, E. Miller, L. Weiss, and P. Campbell. "Cell population tracking and lineage construction with spatiotemporal context". *Medical Image Analysis*, Vol. 12, No. 5, pp. 546–566, October 2008.

[Liaw 02] A. Liaw and M. Wiener. "Classification and regression by randomForest". *R News*, Vol. 2, No. 3, pp. 18–22, December 2002.

[Lind 09] T. Lindeberg. *Scale-Space*, pp. 2495–2504. John Wiley & Sons, Inc., Hoboken, NJ, USA, 2009.

[Lind 98] T. Lindeberg. "Feature detection with automatic scale selection". *International Journal of Computer Vision*, Vol. 30, No. 2, pp. 79–116, November 1998.

[Long 05] X. Long, W. Cleveland, and Y. Yao. "A new preprocessing approach for cell recognition". *IEEE Transactions on Information Technology in Biomedicine*, Vol. 9, No. 3, pp. 407–412, September 2005.

[Long 06] X. Long, W. Cleveland, and Y. Yao. "Automatic detection of unstained viable cells in bright field images using a support vector machine with an improved training procedure". *Computers in Biology and Medicine*, Vol. 36, No. 4, pp. 339–362, April 2006.

[Louk 03] C. G. Loukas, G. D. Wilson, B. Vojnovic, and A. Linney. "An image analysis-based approach for automated counting of cancer cell nuclei in tissue sections". *Cytometry Part A*, Vol. 55A, No. 1, pp. 30–42, September 2003.

[Lowe 04] D. Lowe. "Distinctive image features from scale-invariant keypoints". *International Journal of Computer Vision*, Vol. 60, No. 2, pp. 91–110, November 2004.

[Lowe 99] D. Lowe. "Object recognition from local scale-invariant features". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 1150–1157, Kerkyra, Greece, September 1999.

[Lule 09] V. Lulevich, Y.-P. Shih, S. H. Lo, and G.-Y. Liu. "Cell tracing dyes significantly change single cell mechanics". *The Journal of Physical Chemistry B*, Vol. 113, No. 18, pp. 6511–6519, April 2009.

[Malp 03] N. Malpica, A. Santos, A. Tejedor, A. Torres, M. Castilla, P. García-Barreno, and M. Desco. "Automatic quantification of viability in epithelial cell cultures by texture analysis". *Journal of Microscopy*, Vol. 209, No. 1, pp. 34–40, January 2003.

[Marx 13] V. Marx. "Is super-resolution microscopy right for you?". *Nature Methods*, Vol. 10, No. 12, pp. 1157–1163, December 2013.

[Mell 05] M. Mellor and M. Brady. "Phase mutual information as a similarity measure for registration". *Medical Image Analysis*, Vol. 9, No. 4, pp. 330–343, August 2005.

[Meye 03] D. Meyer, F. Leisch, and K. Hornik. "The support vector machine under test". *Neurocomputing*, Vol. 55, No. 1-2, pp. 169–186, September 2003.

[Mir 12] M. Mir, B. Bhaduri, R. Wang, R. Zhu, and G. Popescu. "Quantitative phase imaging". In: Emil Wolf, Ed., *Progress in Optics*, pp. 133–218, Elsevier, July 2012.

[Mitt 10] S. Mittal, Y. Zheng, B. Georgescu, F. Vega-Higuera, S. Zhou, P. Meer, and D. Comaniciu. "Fast automatic detection of calcified coronary lesions in 3D cardiac CT images". In: *First International Workshop, Machine Learning in Medical Imaging (MLMI), Held in Conjunction with MICCAI 2010*, pp. 1–9, Springer, Beijing, China, September 2010.

[Morg 95] B. Morgan and A. Ray. "Non-uniqueness and inversions in cluster analysis". *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, Vol. 44, No. 1, pp. 117–134, 1995.

[Morr 86] M. C. Morrone, J. Ross, D. C. Burr, and R. Owens. "Mach bands are phase dependent". *Nature*, Vol. 324, No. 6094, pp. 250–253, November 1986.

[Mual 12] F. Mualla, M. Prümmer, D. Hahn, and J. Hornegger. "Toward automatic detection of vessel stenoses in cerebral 3D DSA volumes". *Physics in Medicine and Biology*, Vol. 57, No. 9, pp. 2555–2573, May 2012.

[Mual 13a] F. Mualla, S. Schöll, B. Sommerfeldt, A. Maier, and J. Hornegger. "Automatic cell detection in bright-field microscope images using SIFT, random forests, and hierarchical clustering". *Medical Imaging, IEEE Transactions on*, Vol. 32, No. 12, pp. 2274–2286, December 2013.

[Mual 13b] F. Mualla, S. Schöll, C. Bohr, H. Neumann, and A. Maier. "Epithelial cell detection in endomicroscopy images of the vocal folds". In: E. K. Polychroniadis, A. Y. Oral, and M. Ozer, Eds., *Springer Proceedings in Physics 154, Proceedings of InterM*, pp. 201–205, Antalya, Turkey, October 2013.

[Mual 13c] F. Mualla, S. Schöll, B. Sommerfeldt, and J. Hornegger. "Using the monogenic signal for cell-background classification in bright-field microscope images". In: *Proceedings des Workshops Bildverarbeitung für die Medizin 2013*, pp. 170–174, Heidelberg, Germany, March 2013.

[Mual 14a] F. Mualla, S. Schöll, B. Sommerfeldt, S. Steidl, R. Buchholz, and J. Hornegger. "Improving joint learning of suspended and adherent cell detection using low-pass monogenic phase and transport of intensity equation". In: *Biomedical Imaging (ISBI), 2014 IEEE 11th International Symposium on*, pp. 927–930, Beijing, China, April 2014.

[Mual 14b] F. Mualla, S. Schöll, B. Sommerfeldt, A. Maier, S. Steidl, R. Buchholz, and J. Hornegger. "Unsupervised unstained cell detection by SIFT keypoint clustering and self-labeling algorithm". In: P. Golland, N. Hata, C. Barillot, J. Hornegger, and R. Howe, Eds., *Medical Image Computing and Computer-Assisted Intervention MICCAI 2014*, pp. 377–384, Springer International Publishing, Boston, MA, USA, September 2014.

[Mual 14c] F. Mualla, S. Schöll, B. Sommerfeldt, A. Maier, S. Steidl, R. Buchholz, and J. Hornegger. "Using the low-pass monogenic signal framework for cell/background classification on multiple cell lines in bright-field microscope images". *International Journal of Computer Assisted Radiology and Surgery*, Vol. 9, No. 3, pp. 379–386, May 2014.

[Murp 02] D. B. Murphy. *Fundamentals of Light Microscopy and Electronic Imaging*. John Wiley & Sons, Inc., Hoboken, NJ, USA, 2002.

[Murt 12] F. Murtagh and P. Contreras. "Algorithms for hierarchical clustering: an overview". *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, Vol. 2, No. 1, pp. 86–97, January/February 2012.

[Murt 85] F. Murtagh. "Multidimensional clustering algorithms". *Compstat Lectures, Vienna: Physica Verlag, 1985*, Vol. 4, 1985.

[Natt 99] T. Nattkemper, H. Ritter, and W. Schubert. "Extracting patterns of lymphocyte fluorescence from digital microscope images". In: *Intelligent Data Analysis in Medicine and Pharmacology*, pp. 79–88, Washington, DC, USA, November 1999.

[Opst 94] W. van Opstal, C. Ranger, O. Lejeune, P. Forgez, H. Boudin, J. Bisconte, and W. Rostene. "Automated image analyzing system for the quantitative study of living cells in culture". *Microscopy Research and Technique*, Vol. 28, No. 5, pp. 440–447, August 1994.

[Paga 04] D. Paganin, A. Barty, P. McMahon, and K. Nugent. "Quantitative phase-amplitude microscopy. III. The effects of noise". *Journal of Microscopy*, Vol. 214, No. 1, pp. 51–61, April 2004.

[Paga 98] D. Paganin and K. A. Nugent. "Non-interferometric phase imaging with partially coherent light". *Physical Review Letters*, Vol. 80, No. 12, pp. 2586–2589, March 1998.

[Pan 09] J. Pan, T. Kanade, and M. Chen. "Learning to detect different types of cells under phase contrast microscopy". In: *Microscopic Image Analysis with Applications in Biology (MIAAB)*, Bethesda, MD, USA, September 2009.

[Pan 10] J. Pan, T. Kanade, and M. Chen. "Heterogeneous conditional random field: realizing joint detection and segmentation of cell regions in microscopic images". In: *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*, pp. 2940–2947, San Francisco, CA, USA, June 2010.

[Papa 98] C. Papageorgiou, M. Oren, and T. Poggio. "A general framework for object detection". In: *Sixth International Conference on Computer Vision (ICCV)*, pp. 555–562, Bombay, India, January 1998.

[Pedr 06] F. L. Pedrotti, L. M. Pedrotti, and L. S. Pedrotti. *Introduction to Optics*. Benjamin Cummings, San Francisco, CA, USA, 3 Ed., 2006.

[Plat 99] J. Platt. "Probabilistic outputs for support vector machines and comparisons to regularized likelihood methods". In: *Advances in Large Margin Classifiers*, pp. 61–74, MIT Press, Cambridge, MA, USA, 1999.

[Poul 10] A. D. Poularikas. *Handbook of Formulas and Tables for Signal Processing*. Vol. 13, CRC Press, Boca Raton, FL, USA, 2010.

[Quin 86] J. R. Quinlan. "Induction of decision trees". *Machine Learning*, Vol. 1, No. 1, pp. 81–106, March 1986.

[Quin 93] J. R. Quinlan. *C4.5: Programs for Machine Learning*. Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, 1993.

[Rost 05] E. Rosten and T. Drummond. "Fusing points and lines for high performance tracking". In: *Tenth IEEE International Conference on Computer Vision (ICCV)*, pp. 1508–1515 Vol. 2, Beijing, China, October 2005.

[Rugg 05] A. Ruggeri, E. Grisan, and J. Jaroszewski. "A new system for the automatic estimation of endothelial cell density in donor corneas". *British Journal of Ophthalmology*, Vol. 89, No. 3, p. 306, March 2005.

[Russ 08] B. Russell, A. Torralba, K. Murphy, and W. Freeman. "LabelMe: A database and web-based tool for image annotation". *International Journal of Computer Vision*, Vol. 77, No. 1, pp. 157–173, May 2008.

[Sale 07] B. E. A. Saleh and M. C. Teich. *Fundamentals of Photonics. Wiley series in pure and applied optics*, Wiley, New York, NY, USA, second Ed., 2007.

[Scho 13] S. Schöll, F. Mualla, B. Sommerfeldt, S. Steidl, and A. Maier. "Image preprocessing pipeline for bright-field miniature live cell microscopy prototypes". In: E. K. Polychroniadis, A. Y. Oral, and M. Ozer, Eds., *Springer Proceedings in Physics 154, Proceedings of InterM*, pp. 261–267, Antalya, Turkey, October 2013.

[Scho 14] S. Schöll, F. Mualla, B. Sommerfeldt, S. Steidl, A. Maier, R. Buchholz, and J. Hornegger. "Influence of the phase effect on gradient-based and statistics-based focus measures in bright field microscopy". *Journal of Microscopy*, Vol. 254, No. 2, pp. 65–74, May 2014.

[Sjos 99] P. J. Sjöström, B. R. Frydel, and L. U. Wahlberg. "Artificial neural network-aided image analysis system for cell counting". *Cytometry*, Vol. 36, No. 1, pp. 18–26, May 1999.

[Smit 09] K. Smith, A. Carleton, and V. Lepetit. "Fast ray features for learning irregular shapes". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 397–404, Kyoto, Japan, October 2009.

[Soil 13] P. Soille. *Morphological Image Analysis: Principles and Applications*. Springer Berlin Heidelberg, 2013.

[Soto 03] M. Soto, E. Acosta, and S. Ríos. "Performance analysis of curvature sensors: optimum positioning of the measurement planes". *Opt. Express*, Vol. 11, No. 20, pp. 2577–2588, October 2003.

[Stei 70] E. M. Stein. *Singular Integrals and Differentiability Properties of Functions*. Vol. 2, Princeton University Press, Princeton, NJ, USA, 1970.

[Tadr 10] P. Tadrous. "A method of PSF generation for 3D brightfield deconvolution". *Journal of Microscopy*, Vol. 237, No. 2, pp. 192–199, February 2010.

[Teag 83] M. R. Teague. "Deterministic phase retrieval: a Green's function solution". *Journal of the Optical Society of America*, Vol. 73, No. 11, pp. 1434–1441, November 1983.

[Tsch 08] M. Tscherepanow, F. Zöllner, M. Hillebrand, and F. Kummert. "Automatic segmentation of unstained living cells in bright-field microscope images". In: *Advances in Mass Data Analysis of Images and Signals in Medicine, Biotechnology, Chemistry and Food Industry*, pp. 158–172, Springer, Leipzig, Germany, July 2008.

[Tsoc 04] I. Tsochantaridis, T. Hofmann, T. Joachims, and Y. Altun. "Support vector machine learning for interdependent and structured output spaces". In: *Proceedings of the 21st International Conference on Machine Learning (ICML)*, pp. 104–111, ACM, Banff, Alberta, Canada, July 2004.

[Tu 05] Z. Tu. "Probabilistic boosting-tree: Learning discriminative models for classification, recognition, and clustering". In: *Proceedings of the IEEE International Conference on Computer Vision (ICCV)*, pp. 1589–1596, Beijing, China, October 2005.

[Tuyt 08] T. Tuytelaars and K. Mikolajczyk. "Local invariant feature detectors: a survey". *Foundations and Trends in Computer Graphics and Vision*, Vol. 3, No. 3, pp. 177–280, January 2008.

[Van 13] E. Van Heerden. *Automatic Cell Viability Determination in Bright-Field Microscope Images.* Master's thesis, Pattern Recognition Lab, Erlangen, Germany, 2013. Supervisors: Firas Mualla, Simon Schöll, Joachim Hornegger.

[Veda 08] A. Vedaldi and B. Fulkerson. "VLFeat: An Open and Portable Library of Computer Vision Algorithms". `http://www.vlfeat.org/`, 2008.

[Viol 01] P. Viola and M. Jones. "Rapid object detection using a boosted cascade of simple features". In: *Computer Vision and Pattern Recognition (CVPR) 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, pp. 511–518, Kauai, HI, USA, December 2001.

[Volk 02] V. Volkov, Y. Zhu, and M. De Graef. "A new symmetrized solution for phase retrieval using the transport of intensity equation". *Micron*, Vol. 33, No. 5, pp. 411–416, 2002.

[Wall 10] L. Waller, L. Tian, and G. Barbastathis. "Transport of intensity phase-amplitude imaging with higher order intensity derivatives". *Opt. Express*, Vol. 18, No. 12, pp. 12552–12561, June 2010.

[Wei 08] N. Wei, E. Flaschel, K. Friehs, and T. W. Nattkemper. "A machine vision system for automated non-invasive assessment of cell viability via dark field microscopy, wavelet feature selection and classification". *BMC Bioinformatics*, Vol. 9, No. 1, p. 449, October 2008.

[Weic 98] J. Weickert. *Anisotropic Diffusion in Image Processing.* Teubner, Stuttgart, Germany, 1998.

[Wils 12] S. M. Wilson and A. Bacic. "Preparation of plant cells for transmission electron microscopy to optimize immunogold labeling of carbohydrate and protein epitopes". *Nature Protocols*, Vol. 7, No. 9, pp. 1716–1727, August 2012.

[Wu 08] Q. Wu, F. Merchant, and Castleman. *Microscope Image Processing.* Academic Press, Burlington, MA, USA, 2008.

[Wu 95] K. Wu, D. Gauthier, and M. Levine. "Live cell image segmentation". *IEEE Transactions on Biomedical Engineering*, Vol. 42, No. 1, pp. 1–12, January 1995.

[Yin 10] Z. Yin, R. Bise, M. Chen, and T. Kanade. "Cell segmentation in microscopy imagery using a bag of local Bayesian classifiers". In: *Biomedical Imaging: From Nano to Macro, IEEE International Symposium on*, pp. 125–128, Rotterdam, Netherlands, April 2010.

[Zari 11] A. Zaritsky, S. Natan, J. Horev, I. Hecht, L. Wolf, E. Ben-Jacob, and I. Tsarfaty. "Cell motility dynamics: A novel segmentation algorithm to quantify multi-cellular bright field microscopy images". *PLoS ONE*, Vol. 6, No. 11, p. e27593, November 2011.

[Zern 55] F. Zernike. "How I discovered phase contrast". *Science*, Vol. 121, No. 3141, pp. 345–349, March 1955.