DISSERTATION

submitted to the

Combined Faculty of Mathematics,

Engineering and Natural Sciences

of

Heidelberg University, Germany

for the degree of

Doctor of Natural Sciences

Put forward by

Leonid Kostrykin

born in St. Petersburg, Russia

Globally Optimal Cell Segmentation using Shape and Intensity Information

Advisor: PD Dr. Karl Rohr

Oral examination: .....

Abstract

Studies of cellular structures and processes are of key interest in biomedical research and pathology. Such studies often require segmentation of cell nuclei in microscopy images, for example, for cell counting, analysis of the morphology, or for analysis of other cellular structures in the proximity of nuclei. Since accurate manual segmentation of cell nuclei is tedious, automatic segmentation methods are indispensable to facilitate the analysis. Segmentation of cell microscopy images is particularly challenging due to imaging artifacts like strong image noise and intensity inhomogeneities, but also due to closely clustered or partially overlapping objects and shape variation of cell nuclei.

In this thesis, three new cell segmentation methods are introduced. The methods are based on implicitly parameterized shape models and address major challenges in cell segmentation by jointly exploiting shape and intensity information. Model fitting is performed by energy minimization, comprising convex and combinatorial optimization schemes, which yields results close to global optimality. Convexity is a computationally favorable property which permits fast, robust, and reproducible energy minimization, independently of the initialization.

The proposed cell segmentation methods are based on three new shape parameterizations. First, a non-linear parameterization for elliptical models is presented, which uses the locations of priorly detected objects. Energy minimization is performed by convex optimization using a sequential approximation scheme. Second, a linear parameterization for elliptical models is proposed. This parameterization has the advantage of directly yielding a convex energy, thus sequential approximation is not required. Third, a linear parameterization for deformable shape models is introduced, which also yields a convex energy but permits coping with more general shapes.

To enable joint cell segmentation and cluster splitting, the shape parameterizations are generalized from the single-object to the multi-object case. The corresponding energy is non-convex, yet, we show that it is structurally similar to the min-weight set-cover problem. We develop a novel iterative global energy minimization method which exploits the set-cover structure and provably determines a solution close to global optimality. This is achieved by a new necessary optimality condition, which is iteratively evaluated and refined. In addition, a closed-form solution for non-clustered cell nuclei is derived, which directly determines the corresponding segmentation result and further accelerates the computations.

The proposed methods were applied to challenging image data, comprising fluorescence microscopy images of six different cell types and publicly available benchmark datasets, and a quantitative comparison with previous methods was performed. It turned out that the proposed methods generally yield competitive or improved results. Furthermore, the applicability of the methods to H&E-stained pathology images was investigated.

Zusammenfassung

Studien über zelluläre Strukturen und Prozesse sind von zentralem Interesse in der biomedizinischen Forschung und der Pathologie. Solche Studien erfordern häufig eine Segmentierung der Zellkerne in Mikroskopieaufnahmen, beispielsweise für Zellzählung, Analyse der Morphologie oder für die Analyse anderer zellulärer Strukturen in der Umgebung der Zellkerne. Da eine genaue manuelle Segmentierung der Zellkerne mühsam ist, sind automatische Zellsegmentierungsverfahren zur Durchführung der Bildanalyse unverzichtbar. Besonders schwierig ist die Segmentierung von Zellmikroskopiebildern aufgrund von Bildgebungsartefakten wie starkem Bildrauschen und Intensitätsinhomogenitäten, aber auch aufgrund von zusammenhängenden oder teilweise überlappenden Objekten sowie Formvariationen der Zellkerne.

In dieser Arbeit werden drei neue Methoden zur Zellsegmentierung vorgestellt. Die Methoden basieren auf implizit parametrisierten Formmodellen und begegnen den Herausforderungen der Zellsegmentierung durch die gleichzeitige Nutzung von Form- und Intensitätsinformationen. Die Modellanpassung erfolgt durch Energieminimierung, die konvexe und kombinatorische Optimierungsverfahren umfasst und Ergebnisse nahe der globalen Optimalität liefert. Konvexität ist eine rechnerisch günstige Eigenschaft, die eine schnelle, robuste und reproduzierbare Energieminimierung ermöglicht, unabhängig von der Initialisierung.

Die vorgeschlagenen Methoden basieren auf drei neuen Parametrisierungen der Formmodelle. Zunächst wird eine nicht-lineare Parametrisierung für elliptische Modelle präsentiert, die die Positionen der zuvor detektierten Objekte berücksichtigt. Die Energieminimierung erfolgt durch konvexe Optimierung unter Verwendung eines sequenziellen Approximationsverfahrens. Zweitens wird eine lineare Parametrisierung für elliptische Modelle beschrieben. Diese Parametrisierung hat den Vorteil, dass sie direkt zu einer konvexen Energie führt, so dass eine sequenzielle Approximation nicht erforderlich ist. Drittens wird eine lineare Parametrisierung für deformierbare Formmodelle eingeführt, die ebenfalls zu einer konvexen Energie führt, aber es gestattet mit allgemeineren Formen umzugehen.

Um die Segmentierung und Trennung zusammenhängender Zellen simultan durchführen zu können, werden die Formmodelle für ein einzelnes Objekt auf mehrere Objekte verallgemeinert. Die entsprechende Energie ist nichtkonvex, jedoch zeigen wir, dass sie strukturell dem minimal-gewichteten Mengenüberdeckungsproblem ähnlich ist. Wir entwickeln ein neuartiges iteratives globales Energieminimierungsverfahren, das die Struktur des Mengenüberdeckungsproblems ausnutzt und nachweislich eine Lösung nahe der globalen Optimalität liefert. Dies wird durch eine neue notwendige Optimalitätsbedingung erreicht, die iterativ ausgewertet und verbessert wird. Darüber hinaus wird eine Lösung in analytisch geschlossener Form für nicht-zusammenhängende Zellkerne hergeleitet, die das

entsprechende Segmentierungsergebnis direkt bestimmt und die Berechnungen dadurch weiter beschleunigt.

Die vorgeschlagenen Methoden wurden auf herausfordernde Bilddaten angewendet, die Fluoreszenzmikroskopiebilder von sechs verschiedenen Zelltypen und öffentlich zugänglichen Benchmark-Datensätzen umfassen, und ein quantitativer Vergleich mit früheren Methoden wurde durchgeführt. Es zeigte sich, dass die vorgeschlagenen Methoden im Allgemeinen vergleichbare oder bessere Ergebnisse liefern. Auch wurde die Anwendbarkeit der Methoden auf H&E-gefärbte Pathologiebilder untersucht.

Acknowledgements

I am very grateful to many people who strongly supported me in the creation of this thesis at the Faculty for Mathematics and Computer Science at Heidelberg University. I am also grateful to the whole university for providing a living and stimulating campus environment.

First of all, I would like to thank *PD Dr. Karl Rohr* for supervising my work, providing a pleasant and productive work environment, and the freedom to work on the topics I am most interested in. I am most grateful for his patience and incredibly supporting guidance during preparation of publications.

I would also like to thank my second supervisor *Prof. Dr. Christoph Schnörr* for his introduction to the fascinating topics of convex analysis and optimization, his constant support, and for pointing out research directions.

I am grateful to my former and current colleagues at the Biomedical Computer Vision group for sharing ideas, providing support, and having a pleasant time at social events, as well as the Image and Pattern Analysis group for technical discussions and company during the annual retreats. Special thanks goes to *Christian Ritter* and *Roman Spilger* for being great companions and always keeping the spirit high. I would also like to thank *Christian Ritter* and *Artjom Zern* for countless fruitful discussions. I am also very grateful to the secretaries *Evelyn Wilhelm*, *Barbara Werner*, and *Sabrina Wetzel* for always being available and providing an indispensable level of support in all kinds of bureaucratic matters. I want to thank my supervised interns *Juri Jegelka* and *Sophie Kopeschke* for exhaustive research of literature, and especially *Ronja Rappold* for providing the HeLa dataset, labeling ground truth, helping with the evaluation, and her crucial support in finding affordable housing in Heidelberg.

I am very grateful to my dearest friends *Johannes Heinrichs*, *Alexander Cremer*, and *Helen Baginski* for their immense patience, constant support, encouragement, and relaxing company. I am also especially grateful to my former supervisors *Dr. Matias de la Fuente* (RWTH Aachen) and *Prof. Dr. Stefan Harmeling* (HHU Düsseldorf) for awakening my passion for scientific work, image processing, and computer vision. My deepest gratitude belongs to my parents and my girlfriend *Rosa*.

Finally, I would also like to acknowledge the German Research Foundation (DFG) within the RTG 1653 and the Federal Ministry of Education and Research (BMBF) within the projects de.NBI and CancerTelSys for financial support of my work.

Heidelberg, May 2022 *Leonid Kostrykin*

Contents

List of figures

List of tables

Notation

Mathematical symbols frequently used in this thesis.

$\#A$	Cardinality of the set A
$[n]$	Integer interval $\{1, \ldots, n\}$
[statement]	Iverson brackets, [statement] = {1 if statement is true; 0 else}
$\begin{bmatrix} a_1 & \ldots & a_n \end{bmatrix}$	Row vector with components a_1 to a_n
$(a_1, \ldots, a_n)$	Tuple or column vector with components a_1 to a_n
Diag a	Diagonal matrix with diagonal corresponding to the tuple a
$\mathbb{R}$	The real number line
$\mathbb{R}_+$	The non-negative ray of the real number line
$\mathbb{R}_-$	The non-positive ray of the real number line
$\prec, \preceq, \succ, \succeq$	Partial order with respect to the semidefinite cone
Ω	Set of all image points, $\Omega \subset \mathbb{R}^2$
ω	An image region (set of image points, $\omega \subseteq \Omega$)
$\odot$	Hadamard product
g	Image intensities
x	Image point (tuple of coordinates, $x \in \mathbb{R}^2$)
U	Universe of model activity region fragments (image regions)
X	Model activity region (a set or union of fragments)
$\mathscr{U}$	Prototype set of model activity regions
θ	Polynomial model parameters
ξ	Deformation parameters
ϕ	Convex logistic loss function
P	Polytope (feasible set of a linear program)
$\mathbb{0}_n$	Column vector of zeros with dimension n
$\mathbb{1}_n$	Column vector of ones with dimension n
$C_s(\cdot)$	Zero-level set of a model function $s(\cdot)$
%P	Percentage points

Publications

Major parts of this thesis have been published in peer-reviewed journals and at a peer-reviewed conference.

Peer-reviewed journals

L. Kostrykin, C. Schnörr, and K. Rohr, "Globally optimal segmentation of cell nuclei in fluorescence microscopy images using shape and intensity information," *Medical Image Analysis*, vol. 58, 101536, 2019

L. Kostrykin and K. Rohr, "Superadditivity and convex optimization for globally optimal cell segmentation using deformable shape models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *submitted*, *minor revision*, 2022

Peer-reviewed conference

L. Kostrykin, C. Schnörr, and K. Rohr, "Segmentation of cell nuclei using intensity-based model fitting and sequential convex programming," in *Proc. IEEE International Symposium on Biomedical Imaging (ISBI 2018)*, Washington, D.C., USA, April 4–7, IEEE Piscataway, NJ, 654–657, 2018

Chapter 1

Introduction

1.1 Motivation

For centuries, cell biology has been the basis of biomedical research. Mutations of the cell *nucleus* were identified as a cause of degenerative neuromuscular diseases like muscular dystrophy, cardiomyopathy, disorders of the skin and fat tissue, premature aging, cancer, and other human diseases [1, 2, 3]. Cell nuclei of normal (e.g., non-cancer) cells usually have an elliptical shape. Abnormally shaped and sized cell nuclei have become the most common features for cancer diagnosis [4].

Figure 1.1 shows the simplified structure of a mammalian cell [5]. The cell nucleus is enclosed by the *cytoplasm* and contains strings of *nucleotides*, which are biochemical compounds that encode the genetic material of the cell. Besides nucleotides, the cell nucleus also encloses the *nucleolus*. The nucleolus produces the *ribosomes*, which then migrate to the cytoplasm, where they play a key role in the synthesis of proteins. Non-mammalian cells can have a different structure. For example, in single-cell organisms like bacteria, no distinct nucleus exists and the nucleotides are instead directly embedded into the cytoplasm.

Quantitative assessment of cell nuclei in microscopy images generally requires *segmentation* of cell nuclei, i.e. the identification of image regions corresponding to

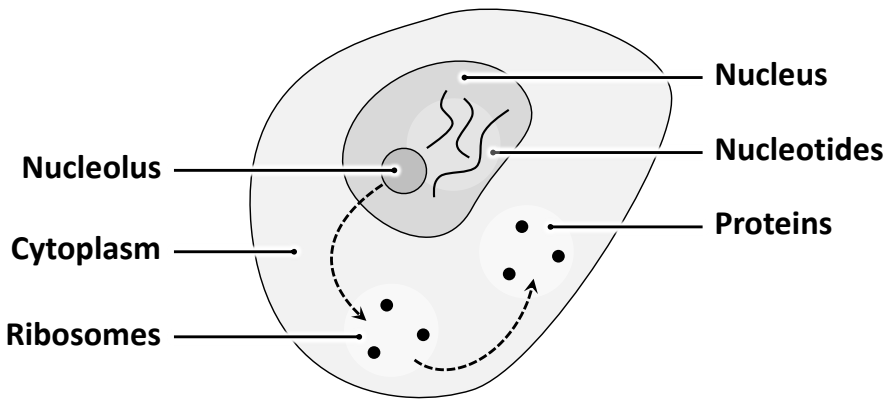
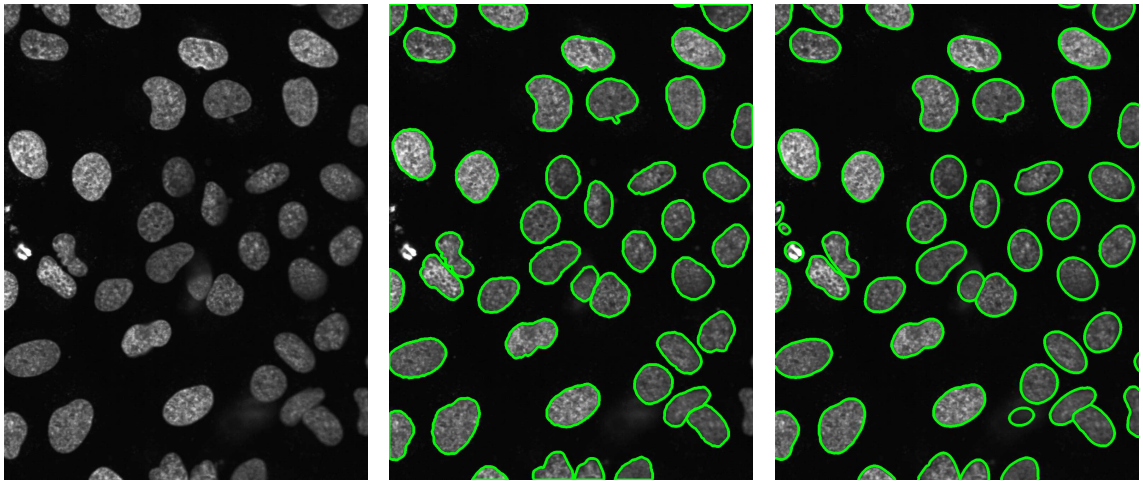


Figure 1.1. Simplified cell structure comprising different cellular components.



(a) (b) (c)

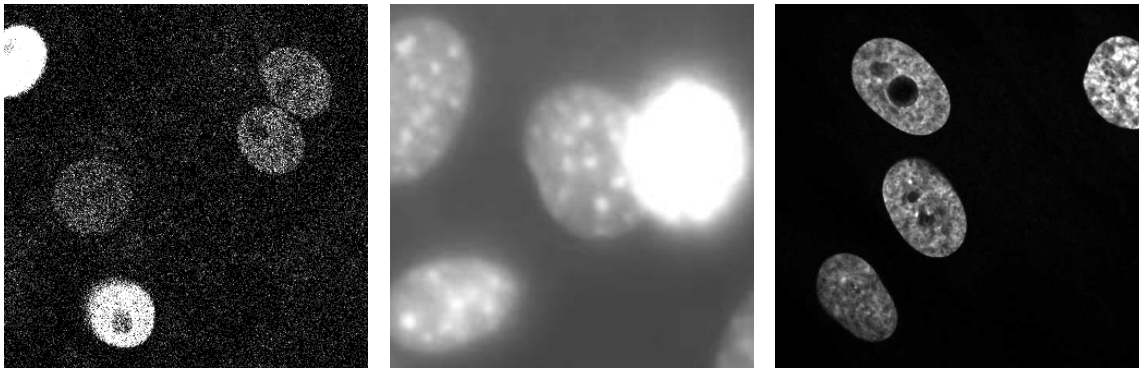
Figure 1.2. Examples of manual and automatic segmentation results (green contours). (a) Microscopy image of U2OS cells. (b) Annotation by a human expert (green contours overlaid with the original image). (c) Automatic segmentation obtained using our approach described in Chapter 6.

individual cell nuclei. Besides the direct analysis of the cell nuclei morphology, cell nuclei segmentation is also important, for example, for cell counting, for analysis of cellular movement or proliferation, but also to determine regions of interest for analysis of other cellular structures. However, manual segmentation is tedious and prone to errors, which has raised interest in methods for *automatic* cell nuclei segmentation. Examples of manual and automatic segmentation results are shown in Figure 1.2. These examples illustrate the major challenges in cell segmentation, such as irregularity of the shapes and closely clustered or partially overlapping objects. Further challenges include imaging artifacts like image noise and intensity inhomogeneities (see Figure 1.3). Such inhomogeneities occur across multiple objects (inter-object inhomogeneities) as well as within individual cell nuclei (intra-object inhomogeneities).

Below, we describe the acquisition process of cell microscopy images and examine the challenges in cell segmentation in more detail.

1.2 Cell microscopy imaging

Human cells are typically around 0.01 to 0.1 mm in size [5] and thus too small to be seen with the naked eye. *Microscopy*, a combination of the Greek terms *mikros* (tiny) and *skopein* (view or observe), comprises the techniques for observation and imaging of such tiny structures. Below, we describe some fundamentals of cell microscopy imaging (for a more comprehensive description see the textbook [6]).



(a) (b) (c)

Figure 1.3. Examples of challenges in cell microscopy segmentation (image sections). (a) Closely clustered cell nuclei and strong image noise. (b) Partially overlapping objects and strong inter-object intensity inhomogeneities. (c) Strong intra-object intensity inhomogeneities.

1.2.1 Optical microscopy and image acquisition

Visible light behaves like an electromagnetic wave with a characteristic wave length. *Refraction* is the well-known optical law, that light changes the direction of propagation when traversing from one optical medium to another, depending on the angle of incidence. In *optical* microscopy, this law is exploited by objectives (groups of lenses) to achieve a magnified *projection* of the specimen. Disjoint objects like the individual cells may overlap within the projection.

Image acquisition is performed, for example, by charge-coupled devices (CCD), which measure the accumulated light intensity in each image pixel of the magnified projection by counting the number of arriving photons over a short period of time. This number of photons is generally Poisson-distributed and induces *Poisson noise* in the image [7]. In addition, the readout of the pixel-wise intensities from the CCD is Gaussian-distributed, leading to additional *Gaussian noise* in the image [7].

In *bright-field* microscopy, which is illustrated in Figure 1.4a, a white background light is used for illumination of the specimen. Differences in the amount of light absorbed by the specimen lead to differences in intensity, which we perceive as *structure*. Due to the tininess of the cellular components, almost no light is absorbed by the specimen, which makes direct optical observation of cellular structures difficult. To cope with that, chemical compounds called *stains* are added to the specimen, which bind to specific cellular components and increase the absorption for characteristic wavelengths of light, thus *staining* the corresponding cellular components in distinctive colors. Hematoxylin is the most widely used stain for cell nuclei. It binds to negatively charged compounds like the nucleotides (cf. Figure 1.1), staining nuclei in blue or purple tones. Another widely used stain is eosin, which binds to the positively charged proteins within the cytoplasm, staining cytoplasms in red. Using hematoxylin in conjunction with eosin (H&E) thus improves the

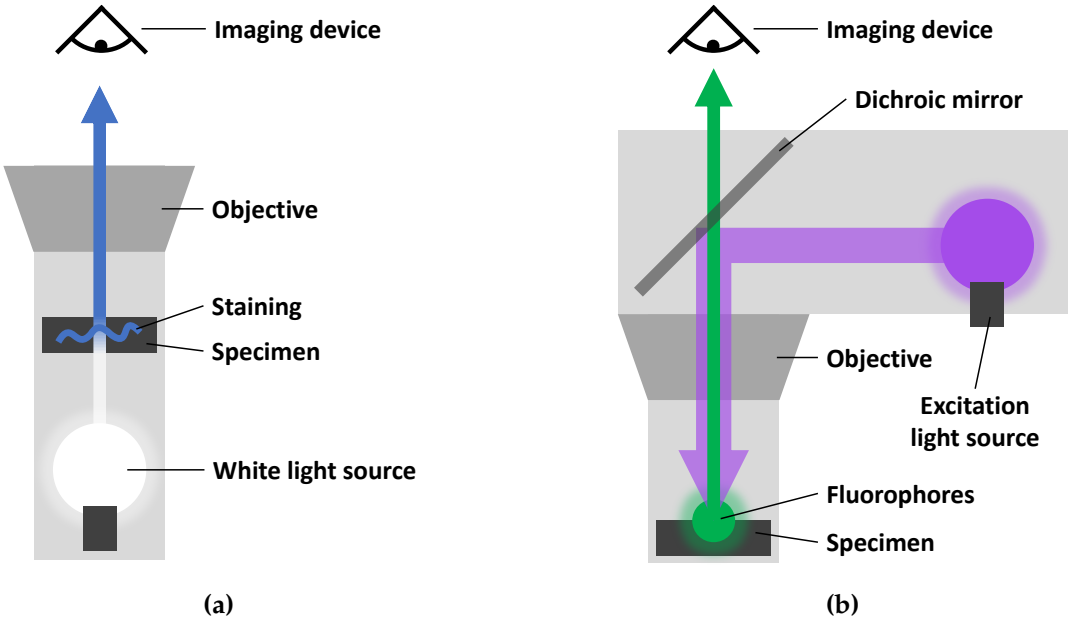


Figure 1.4. Simplified setup of a (a) bright-field and (b) fluorescence microscope. Arrows indicate the corresponding light paths.

optical contrast of the adjacent structures (nucleus and cytoplasm). H&E is the most important staining method in histology.

1.2.2 Fluorescence microscopy

Another type of optical microscopy is *fluorescence* microscopy, which permits very accurate imaging of cellular structures and is one of the most important methods for cell microscopy today. This method requires that fluorescent stains called *fluorophores* are added to the specimen. Different fluorophores bind to different types of cellular structures (e.g., cell nuclei, cytoplasms, mitochondria). Upon light excitation, the fluorophores emit light of characteristic wavelengths, rendering cellular structures visible in distinctive colors. The light path of a fluorescence microscope is illustrated in Figure 1.4b. A dichroic mirror, which is translucent for light of specific wavelengths, is used to split excitation and fluorescence light.

The fluorescence light intensities can be captured in different image channels based on the different colors, and each image channel can be considered as a separate grayscale image. Cell nuclei are usually captured in a single image channel where they appear as bright image regions (high intensities). Since the fluorophores emit light of low intensity, fluorescence microscopy images generally demonstrate comparably strong Poisson noise (e.g., [8, 9]).

DAPI (Diamidino-phenylindole-dihydrochloride) is an often used fluorophore for cell nuclei staining in fluorescence microscopy. DAPI binds strongly to specific parts of the nucleotide molecule (adenine and thymine). Since these parts are not

uniformly distributed within the nucleus, the light emission is inhomogeneous. Moreover, the concentration of nucleotides within the nucleoli (cf. Figure 1.1) is three times lower than within the rest of the nucleus. Hence, less light is emitted from the nucleoli which generally appear as a dark region inside the nucleus [10] (see, e.g., the upper-left nucleus in Figure 1.3c). Hoechst 33342 and the green fluorescent protein (GFP) are other fluorophores, which are often used to label sub-cellular structures and better suited for *in vivo* studies.

Autofluorescence artifacts are imaging artifacts arising specifically in fluorescence microscopy which increase the difficulty of cell segmentation. These artifacts usually correspond to unintended emission of light and can be caused by biological processes or contamination of the specimen. Such contamination can be caused, for example, during preparation of the specimen by chemical compounds which are added in order to preserve cellular structures from decay.

1.3 Contributions and overview

In this thesis, we propose different new methods for globally optimal cell nuclei segmentation, which jointly exploit shape and intensity information. Our methods are based on *(i)* implicitly parameterized shape models, *(ii)* model fitting by global energy minimization, and *(iii)* convex optimization. The methods do not suffer from local energy minima and yield the global solution independently of the initialization. The main contributions of this thesis are:

- **CVXELL, a convex model-based approach for the segmentation of cell nuclei using elliptical models.** We introduce an approach based on *implicitly* parameterized shape models, which are directly fitted to the image intensities. Implicitly parameterized shape models were previously not used for cell segmentation. Our approach relies on elliptical models and a sequential approximation scheme, which yields a sequence of *convex* programs. Convex optimization has the advantages that it is fast, robust, and yields globally optimal results independently of the initialization. We use an efficient second-order optimization scheme to solve the sequence of convex programs and estimate the *globally optimal* solution based on the image intensities. Model fitting is performed within image regions, which correspond to object detections and are determined by exploiting the local image structure. The approach is denoted CVXELL since it is based on convexity and elliptical models. We evaluated the approach using fluorescence microscopy images of two different cell types and performed a quantitative comparison with previous methods. The work was published in [11].

- **GOCELL, a globally optimal approach for joint cell segmentation and cluster splitting using elliptical models.** We introduce the approach GOCELL

(globally optimal collaborating ellipses), which is based on a novel implicit shape parameterization for elliptical objects. In the *single-object* case, this parameterization leads to *convex* energies which can be directly minimized without requiring an approximation. In the *multi-object* case, multiple collaborating ellipses are used, which has the advantage that prior detection of individual cell nuclei is not needed. This leads to a non-convex energy, yet we have found that model fitting using the multi-object model corresponds to the combinatorial *min-weight set-cover* problem. The result is determined close to the *globally optimal* solution using an efficient combination of combinatorial and second-order convex optimization schemes. The proposed approach *jointly* performs cell segmentation and cluster splitting and naturally copes with touching and partially overlapping cell nuclei, is robust, and computationally efficient. In contrast, previous shape-based approaches for cell segmentation either are computationally intractable, not globally optimal, require prior image binarization, or object detection. We successfully applied our approach to fluorescence microscopy images of five different cell types and performed a quantitative comparison with previous methods. The work was published in [12].

- **SuperDSM, a globally optimal approach based on superadditivity and deformable shape models for joint cell nuclei segmentation and cluster splitting.** We introduce the approach SuperDSM (superadditivity and deformable shape models), which naturally copes with *deformable* shapes. Previous methods are limited to elliptical models, not globally optimal, or computationally intractable. We propose an implicit parameterization of deformable shape models and show that it leads to a *convex* energy. To *jointly* perform cell nuclei segmentation and cluster splitting, we developed a novel iterative global energy minimization method, which solves the corresponding combinatorial min-weight set-cover problem by leveraging the inherent property of *superadditivity* of the energy. This property exploits the lower bound of the energy of the union of the models and establishes a *necessary optimality condition* for the set-cover, which improves the computational efficiency. Our method provably determines a solution close to global optimality. In addition, we derive a *closed-form solution* of the min-weight set-cover problem based on the superadditivity property for non-clustered cell nuclei, which further reduces computational cost. We evaluated our method using fluorescence microscopy images of five different cell types comprising various challenges, and performed a quantitative comparison with previous methods. Our method achieved state-of-the-art or improved performance. Furthermore, we successfully applied the method for segmentation of H&E-stained pathology images. The work has been submitted for publication [13].

This thesis is organized as follows. In Chapter 2, we describe the mathematical background of optimization with focus on convex optimization, which is the basis for the proposed methods. We also review previous work on cell nuclei segmentation. In Chapter 3, we describe the datasets, baseline methods, and performance measures used for evaluation. In Chapter 4, we present our CVXELL approach based on elliptical models, convex optimization, and sequential approximation. In Chapter 5, we introduce a novel implicit parameterization for elliptical models. This parameterization directly yields convex energies in the single-object case, therefore sequential approximation is not required. We generalize this parameterization to the multi-object case and describe our GOCELL approach, which jointly performs cell segmentation and cluster splitting, so prior object detection also is not required. In Chapter 6, we introduce our SuperDSM approach, which naturally copes with shape variation using deformable shape models. We also propose a novel optimization scheme, which leverages the property of superadditivity, so that SuperDSM is both more efficient and more general than GOCELL. Finally, in Chapter 7, we provide a summary of our work and describe future research directions.

An overview of the developed methods, their components, and their interrelations is given in Figure 1.5. The single-object energy is the central building block of the three methods. For CVXELL, this energy is based on a non-linear parameterization for elliptical shape models and a convex loss function. For GOCELL, a linear shape parameterization is used instead, and a generalization to the multi-object case is proposed. This corresponds to a min-weight set-cover problem, which incorporates multiple convex optimization problems corresponding to the single-object energy. For SuperDSM, the single-object energy is based on a linear parameterization for deformable shape models, and we introduce a regularization scheme for the deformations. The multi-object case is solved more efficiently by automatically confining the computations to a meaningful subset of the convex optimization problems for the single-object energies while GOCELL needs to process the whole set.

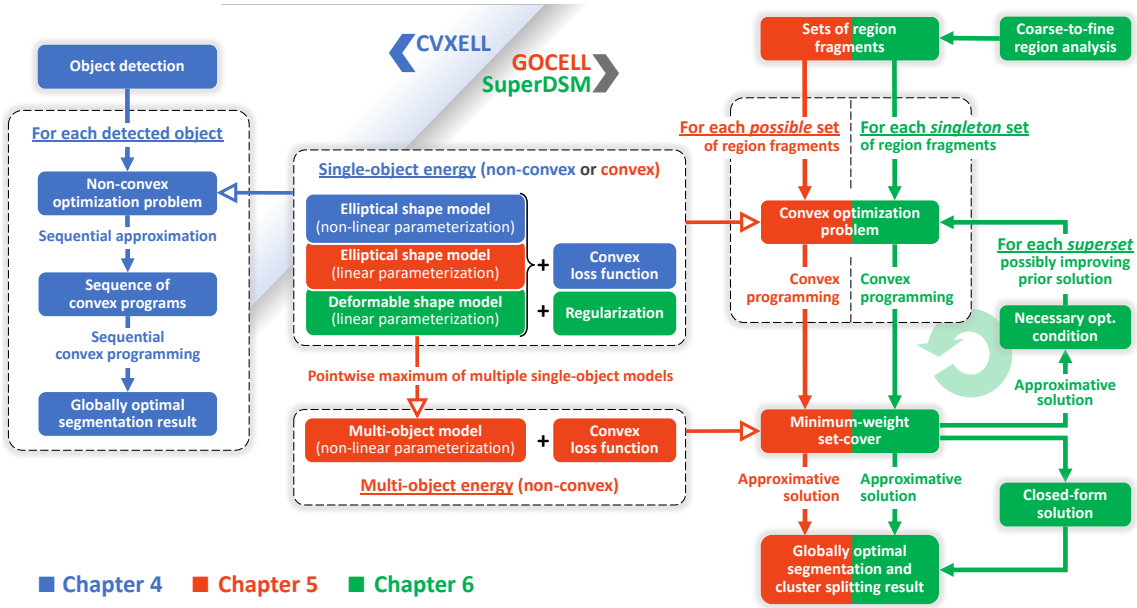


Figure 1.5. The interrelations of the chapters in this thesis. Hollow arrow heads indicate conceptual relations. Solid arrow heads indicate the main algorithmic steps. Dashed boxes indicate groups of algorithmic steps or components.

Chapter 2

Foundations and previous work

2.1 Introduction

This chapter describes the foundations of our work. Many approaches for cell nuclei segmentation, including the approaches proposed in this thesis, are based on energy minimization. Therefore, we first introduce the fundamental concepts of optimization (Section 2.2). We then present an overview of previous work on cell segmentation (Section 2.3).

2.2 Fundamentals of optimization

In this section, we introduce the fundamental concepts and technical background of optimization. Below, we first summarize basic canonical definitions based on the textbooks [14, 15, 16].

Optimization means the process of solving an *optimization problem*. Optimization problems are either minimization or maximization problems. In the most general form, a minimization problem is defined by

$$\begin{aligned} &\text{minimize} && f_0(x) \\ &\text{subject to} && x \in C, \end{aligned} \tag{2.1a}$$

where $x \in \mathbb{R}^n$ is the vector to be optimized (e.g., the parameters of a model), the function $f_0 \colon \mathbb{R}^n \to \mathbb{R}$ is the *objective function* and the set $C \subseteq \mathbb{R}^n$ is the *feasible set* or *feasible region* of the problem. To study optimization problems, it is often useful to use a different notation of the minimization problem in Eq. (2.1a):

$$\inf_{x \in C} f_0(x) \tag{2.1b}$$

This has the advantage that the optimization problem is represented as an analytical term (the infimum term). Maximization problems like $\sup_{x \in C} f_0'(x)$ can be transformed into the form of Eq. (2.1b) by substituting the objective function f_0 by $-f_0'$. Thus, Eq. (2.1b) represents both minimization and maximization problems.

Any vector $\hat{x} \in \mathbb{R}^n$ satisfying the condition $\hat{x} \in C$ is *feasible* or a *feasible solution* of the optimization problem. Any vector $\hat{x} \in \mathbb{R}^n$ satisfying the condition

$$\hat{x} \in C \quad \wedge \quad f_0(\hat{x}) \leq f_0(x) \text{ for all } x \in C \tag{2.2}$$

is an *optimal solution* or a *globally optimal solution* of the optimization problem, and $f_0(\hat{x})$ is the *optimal value* of the problem. An optimal solution exists if and only if the infimum of the objective function f_0 on C is attained, and then $\inf_{x \in C} f_0(x) = \min_{x \in C} f_0(x)$. Determining a globally optimal solution is also called *global optimization*, to better distinguish from local optimization (see below).

A feasible solution $\hat{x} \in C$ is a *locally optimal solution* if the optimality condition in Eq. (2.2) holds locally, i.e. for all $x \in C$ within a neighborhood $\|x - \hat{x}\| \leq \epsilon$ of $\hat{x}$ with arbitrarily small $\epsilon > 0$. A locally optimal solution can be (globally) optimal, but, in general, it is not. Determining a locally optimal solution is referred to as *local optimization* or *non-global optimization*.

An alternative notation of the optimization problem in Eq. (2.1) is to implicitly encode the feasible set C into the objective function. To this end, the *extended real number line* $\overline{\mathbb{R}} = \mathbb{R} \cup \{-\infty, +\infty\}$ and the *indicator function* $\delta_C \colon \mathbb{R}^n \to \overline{\mathbb{R}}$,

$$\delta_C(x) = \begin{cases} 0 & \text{if } x \in C \\ +\infty & \text{else} \end{cases} \tag{2.3}$$

are defined. The minimization problem from Eq. (2.1) can then be written $\inf_{x \in \mathbb{R}^n} f(x)$ with $f = f_0 + \delta_C$. The set $\operatorname{dom} f = \{x \in \mathbb{R}^n | f(x) < \infty\}$ is the *effective domain* of f. Analogously, maximization problems are denoted $\sup_{x \in \mathbb{R}^n} f_0(x) - \delta_C(x)$.

Below, we first discuss aspects of computational cost and tractability of optimization (Section 2.2.1). We then briefly introduce *convexity*, which characterizes a major class of computationally tractable optimization problems (Section 2.2.2). Conjugate and Lagrange duality are then described (Section 2.2.3), the latter providing the foundations for necessary and sufficient optimality conditions (Section 2.2.4). These conditions are directly exploited by interior point methods for convex optimization (Section 2.2.5). Finally, optimization methods for computationally intractable combinatorial problems based on approximations are described (Section 2.2.6).

2.2.1 Computational tractability

The computational cost of computational problems such as optimization or decision problems is commonly assessed based on the *complexity* class of the problems. In this section, we give a short and informal overview of the key concepts of complexity regarding *tractability* of computational problems.

The fundamental complexity classes **P** and **NP** are established for *decision problems* (e.g., [17]). Roughly speaking, a decision problem is the problem of determining whether, for a fixed *input*, a statement is true or false. For example, let $\hat{f}_0 \in \mathbb{R}$ be an

arbitrary value. Then, the optimization problem in Eq. (2.1a) is naturally associated with the decision problem

$$\text{“Given a value } \hat{f}_0 \in \mathbb{R}\text{, is there an } x \in C \text{ so that } f_0(x) \leq \hat{f}_0\text{?”} \tag{2.4}$$

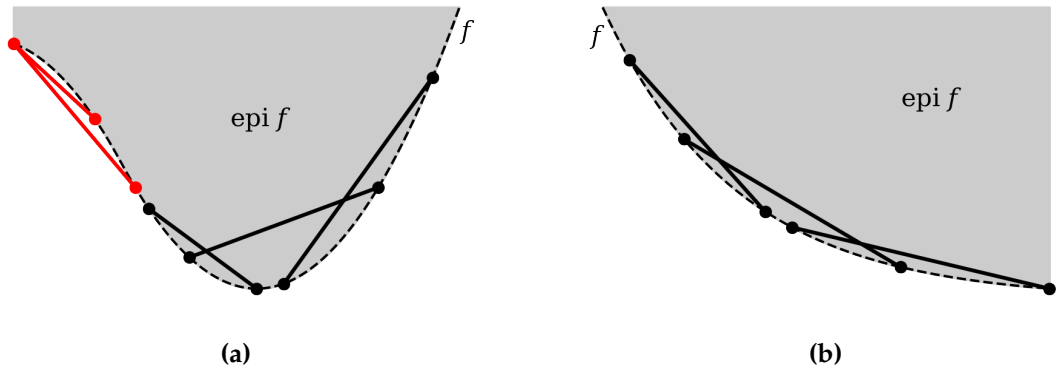
In this example, the vector $x \in C$ is called a *certificate*. The input consists of the value $\hat{f}_0$, the feasible set C, and the objective function f_0. Complexity theory also requires that we assume an *encoding* for $\hat{f}_0$ (e.g., a specific binary representation) as well as f_0 and C (e.g., polynomials of sufficiently high degree and level sets thereof) to properly quantify the *problem size* (e.g., the number of bits of the binary representation, the number n of variables, and the degree of the polynomials used for the encoding).

The complexity class **P** contains decision problems which can be *solved* in polynomial time with respect to the problem size. On the other hand, if a certificate can be obtained by *guessing* and it can be *verified* in polynomial time, then this decision problem is in **NP**. This implies that for any problem in **NP**, the certificate must be polynomial-length, since otherwise it cannot be verified in polynomial time (e.g., $\sqrt{2}$ is of infinite length). It is evident that $\mathbf{P} \subseteq \mathbf{NP}$.

Solving the optimization problem in Eq. (2.1a) obviously also solves the associated decision problem (2.4). Thus, the optimization problem is at least as hard as the associated decision problem. The intuition behind **P** and **NP** is that problems in **P** are “computationally easy” to *solve*, whereas for problems in **NP** solutions are only “computationally easy” to *verify*. Solving problems in $\mathbf{NP} \setminus \mathbf{P}$ is difficult, since *guessing* a polynomial-length certificate amounts to exponential-time trial-and-error schemes when implemented on traditional computers (e.g., brute-force search). Even for small problems with a few tens of variables, exponential run time can amount to hours or days. Problems which are not in **NP** are even more difficult and can exhibit super-exponential run time. Cobham [18] and Edmonds [19] coined the widely accepted rule of thumb, that a computational problem is only *tractable* if it is in **P** (exceptions exist, e.g., due to constant factors).

An optimization or decision problem is **NP**-hard, if it generalizes any problem in **NP** and the generalization can be obtained in polynomial time (e.g., [17]). Intuitively, **NP**-hard problems are thus computationally *at least* as difficult as any problem in **NP**. If the set C in the decision problem (2.4) is encoded as a union of finitely many polynomial level sets, then it corresponds to the decision problem for the *existential theory of the reals* (see, e.g., [20]), which is known to be **NP**-hard [21]. Under the widely believed conjecture that $\mathbf{P} \neq \mathbf{NP}$, this is the reason that optimization is generally difficult or even intractable, unless computationally favorable properties of the feasible set C and the objective function f_0 can be exploited.

A prominent computationally favorable property of optimization problems is *convexity*. Below, we first briefly describe the tools required for analysis of convexity and to tailor *convex* optimization problems (Section 2.2.2). We then describe the



(a) (b)

Figure 2.1. Examples of a (a) non-convex function f and a (b) convex function f (dashed line). The red line segments lie outside of epi f, thus, neither the set epi f nor the function f in (a) are convex. The black line segments lie in epi f.

computational schemes, which perform convex optimization in polynomial time (Section 2.2.5), suggesting that convex optimization problems are associated with decision problems in **P** and thus computationally easy. Finally, we describe a class of optimization problems, which are **NP**-hard, but for which fast and accurate approximation algorithms are known (Section 2.2.6).

2.2.2 Convexity

A set $C \in \mathbb{R}^n$ is *convex* if and only if the straight line segment between any two points of the set lies in C, i.e. $\lambda \cdot x_1 + (1-\lambda) \cdot x_2 \in C$ for all $x_1, x_2 \in C$ and $\lambda \in [0,1]$ (e.g., [14, 15, 16]). A function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$ is convex if and only if the *epigraph* of the function,

$$\operatorname{epi} f = \{(x, \mu) | x \in \operatorname{dom} f, \mu \geq f(x)\}, \tag{2.5}$$

is a convex set (e.g., [14, 15]). Note that convexity of f implies convexity of dom f. Examples are shown in Figure 2.1.

We briefly mention practically useful conditions for convexity of functions $f : \mathbb{R}^n \to \overline{\mathbb{R}}$, which are both necessary and sufficient:

First-order condition. Let f be differentiable on a convex set $\mathcal{X} \subseteq \mathbb{R}^n$ which contains dom f and $\nabla f(x)$ denote the gradient of f at x. Then, f is convex if and only if $f(x_2) \geq f(x_1) + \nabla f(x_1) \cdot (x_1 - x_2)$ for all $x_1, x_2 \in \mathcal{X}$ (see Proposition 1.1.7 in [16] for a proof).

Second-order condition. Let f be twice differentiable on a convex set $\mathcal{X} \subseteq \mathbb{R}^n$ which contains dom f. Then, f is convex if and only if the Hessian matrix $\nabla^2 f(x)$ is positive semidefinite for all $x \in \mathcal{X}$ (see Proposition 1.1.10 in [16] for a proof). This means that the curvature of f is non-negative.

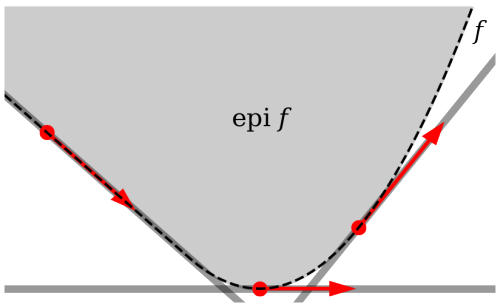


Figure 2.2. Examples of a convex function f (dashed line) and the first-order condition $f(x_2) \geq f(x_1) + \nabla f(x_1) \cdot (x_1 - x_2)$. The red dots are examples of vectors x_1 and the red arrows indicate the corresponding gradients $\nabla f(x_1)$. The gray lines indicate the hyperplanes, which correspond to the rhs of the first-order condition.

Note that differentiability of $f = f_0 + \delta_C$ with $f_0 \colon \mathbb{R}^n \to \mathbb{R}$ means that $C = \mathbb{R}^n$ and $f = f_0$. Hence, the first-order condition directly implies that if f is differentiable and convex, then any *stationary point*, that is a point $\hat{x} \in \mathbb{R}^n$ satisfying $\nabla f(\hat{x}) = 0$, also satisfies Eq. (2.2) and is thus an optimal solution of $\inf_{x \in \mathbb{R}^n} f(x)$. This relation is illustrated in the example in Figure 2.2, where the bottom red dot corresponds to a stationary point. For non-differentiable functions, an analogous relation can be shown using the subgradient of f (see, e.g., Section 5.4.3 in [16]). Thus, if any convex function f has a stationary point $\hat{x}$, then it is a globally optimal solution of $\inf_{x \in \mathbb{R}^n} f(x)$. There are no non-global minima or saddle points in convex functions.

Example 2.1 (Logistic loss). Consider $\phi(y; s) = \log(1 + \exp(-y \cdot s))$, which is the *logistic loss* function (e.g., [22, 23]), as a function of $s \in \mathbb{R}$ for a fixed $y \in \mathbb{R}$. The first derivative is $\frac{\partial}{\partial s}\phi(y; s) = -y/(1 + \exp(y \cdot s))$ and yields the second derivative

$$\frac{\partial^2}{\partial s^2}\phi(y; s) = y^2 \cdot \frac{\exp(y \cdot s)}{(1 + \exp(y \cdot s))^2} \geq 0, \tag{2.6}$$

which is non-negative since it is a product of non-negative terms. Thus, by the necessary and sufficient second-order condition for convexity, the logistic loss function ϕ is *convex* as a function of s. The reasoning applies analogously for ϕ as a function of y for a fixed s, which, however, is not of interest for us.

We briefly mention further rules which are useful for establishing convexity:

Affine functions. Any affine function is convex (the Hessian matrix is zero and thus positive semidefinite).

Affine weighted sum. Let $f_1, \ldots, f_m \colon \mathbb{R}^n \to \mathbb{R}$ be affine functions, associated with weights $w_1, \ldots, w_m \in \mathbb{R}$. Then, the weighted sum $w_1 \cdot f_1 + \cdots + w_m \cdot f_m$ is a convex function (any affine function is convex).

Convex weighted sum. Let $f_1, \ldots, f_m : \mathbb{R}^n \to \overline{\mathbb{R}}$ be convex functions, associated with *non-negative* weights $w_1, \ldots, w_m \in \mathbb{R}_+$. Then, the non-negative weighted sum $w_1 \cdot f_1 + \cdots + w_m \cdot f_m$ is a convex function (the non-negative weighted sum of positive semidefinite Hessian matrices is positive semidefinite).

Convex-affine composition. Let $f_1 : \mathbb{R}^m \to \overline{\mathbb{R}}$ be a convex function and $f_2 : \mathbb{R}^n \to \mathbb{R}^m$ an affine function. Then, the composition $f_1 \circ f_2$ is convex (see Proposition 1.1.4 in [16] for a proof using a linear function f_2, generalization to the affine case is straight-forward).

Pointwise supremum. Let $\mathscr{F}$ be a family of convex functions $f : \mathbb{R}^n \to \overline{\mathbb{R}}$, $f \in \mathscr{F}$. Then, the pointwise supremum $\sup_{f \in \mathscr{F}} f(x)$ is also convex (see Proposition 1.1.6 in [16] for a proof).

An optimization problem of the form in Eq. (2.1) is convex if the feasible set C and the objective function f_0 are convex (e.g., [15]). Below, we give two important examples of convex optimization problems (see Sections 1.4 in [22] and 7.1.1 in [15] for details).

Example 2.2 (Linear regression). Let $x^{(1)}, \ldots, x^{(m)} \in \mathbb{R}^D$ be a sequence of inputs and $y^{(1)}, \ldots, y^{(m)} \in \mathbb{R}$ a sequence of corresponding outputs. In *linear regression*, each output $y^{(i)}$ is modeled by $s\left(x^{(i)}; \theta\right)$ using the model function $s(x; \theta) = \langle \theta, \Phi(x) \rangle$, which is linear in the model parameters $\theta \in \mathbb{R}^n$ and where $\langle \theta, \Phi(x) \rangle$ is the inner product of θ and $\Phi(x)$. The function $\Phi : \mathbb{R}^D \to \mathbb{R}^n$ is called *basis function expansion* or *feature map*. Model fitting is performed by likelihood maximization, assuming that $y^{(i)} \sim p\left(y \middle| x^{(i)}\right)$, where $p(y|x) = \mathcal{N}\left(y \middle| s(x; \theta), \sigma^2\right)$ is the normal distribution with expected value $s(x; \theta)$ and standard deviation $\sigma > 0$. Since log is monotonously increasing, likelihood maximization boils down to

$$\sup_{\theta \in \mathbb{R}^n} \log \prod_{i \in [m]} p\left(y^{(i)} \middle| x^{(i)}\right) = \frac{-1}{2\sigma^2} \cdot \inf_{\theta \in \mathbb{R}^n} \underbrace{\sum_{i \in [m]} \left(y^{(i)} - s\left(x^{(i)}; \theta\right)\right)^2}_{f_0(\theta)}. \tag{2.7}$$

The optimal solution $\hat{\theta}$ of Eq. (2.7) is thus determined by minimization of the objective function f_0 with respect to $\theta \in \mathbb{R}^n$. The rhs of Eq. (2.7) has the form of a *linear least squares* minimization problem, where the residual term $y^{(i)} - s\left(x^{(i)}; \theta\right)$ is linear in θ. Thus, each summand corresponds to a convex-affine composition. Thus, the objective function f_0 is a sum of convex functions, which is *convex* too (by the rule for convex weighted sums, using $w_i = 1$ for $i = 1, \ldots, m$). Since the objective function f_0 and the feasible set $\mathbb{R}^n$ are convex, Eq. (2.7) is a convex problem.

Example 2.3 (Logistic regression). Let $x^{(1)}, \ldots, x^{(m)} \in \mathbb{R}^D$ be a sequence of inputs and $y^{(1)}, \ldots, y^{(m)} \in \{-1, +1\}$ a sequence of corresponding *binary* outputs. In *logistic regression*, a linear model function $s(x; \theta) = \langle \theta, \Phi(x) \rangle$ is used analogously to linear regression but a sigmoid distribution $p(y|x) = 1/(1 + \exp(-y \cdot s(x; \theta)))$ is assumed instead of the normal distribution. Applying the log-likelihood maximization approach analogously to Example 2.2 yields

$$\sup_{\theta \in \mathbb{R}^n} \log \prod_{i \in [m]} p\left(y^{(i)} \middle| x^{(i)}\right) = - \inf_{\theta \in \mathbb{R}^n} \sum_{i \in [m]} \phi\left(y^{(i)}; s\left(x^{(i)}; \theta\right)\right), \tag{2.8}$$

where $\phi(y; s) = \log(1 + \exp(-y \cdot s))$ is the *convex* logistic loss function from Example 2.1. Reasoning analogously to Example 2.2 (in particular, noting that s is affine in θ), it is thus seen that logistic regression is a convex problem.

2.2.3 Duality

We briefly sketch Lagrange duality, which is fundamental for the subsequent sections. Lagrange duality is a result of *convex conjugates*. Given a function $f : \mathbb{R}^n \to \overline{\mathbb{R}}$, the convex conjugate $f^* : \mathbb{R}^n \to \overline{\mathbb{R}}$ of f is defined

$$f^*(p) = \sup_{x \in \mathbb{R}^n} \langle x, p \rangle - f(x), \tag{2.9}$$

which is a function of the vector $p \in \mathbb{R}^n$ and where $\langle x, p \rangle$ is the inner product of the vectors x and p (e.g., [14, 15, 16]). The convex conjugate f^* is a convex function even if f is not convex, since Eq. (2.9) is the pointwise supremum of functions $\langle x, p \rangle - f(x)$, which are affine in p for any fixed $x \in \mathbb{R}^n$ (cf. Section 2.2.2). The definition of the convex conjugate directly implies *Fenchel's inequality*, that is $f^*(p) \geq \langle x, p \rangle - f(x)$ for all $x, p \in \mathbb{R}^n$, or equivalently, $f(x) \geq \langle x, p \rangle - f^*(p)$. Considering the *biconjugate* $f^{**}(x) = \sup_{p \in \mathbb{R}^n} \langle x, p \rangle - f^*(p)$, which is naturally defined as the convex conjugate of f^*, we thus obtain $f \geq f^{**}$.

An intuitive view of the convex conjugate $f^*(p)$ is to consider the hyperplane with normal vector $(-p, 1)$ which is tangential to the graph of f, as illustrated in Figure 2.3a. In this view, the value $-f^*(p)$ corresponds to the vertical intercept of the hyperplane. The convex biconjugate $f^{**}(x)$ is shown in Figure 2.3b and the relation $f \geq f^{**}$ becomes evident from comparison of the graph of f^{**} to epi f. Since the set epi f^{**} corresponds to the *convex hull* of the set epi f, the biconjugate f^{**} is also called the *(lower) convex envelope* or *convex relaxation* of f (e.g., [15]).

Primal and dual problems

We briefly sketch below, that Lagrange duality is easily obtained as a special case of convex conjugates (see, e.g., [16] for details). Consider an auxiliary function

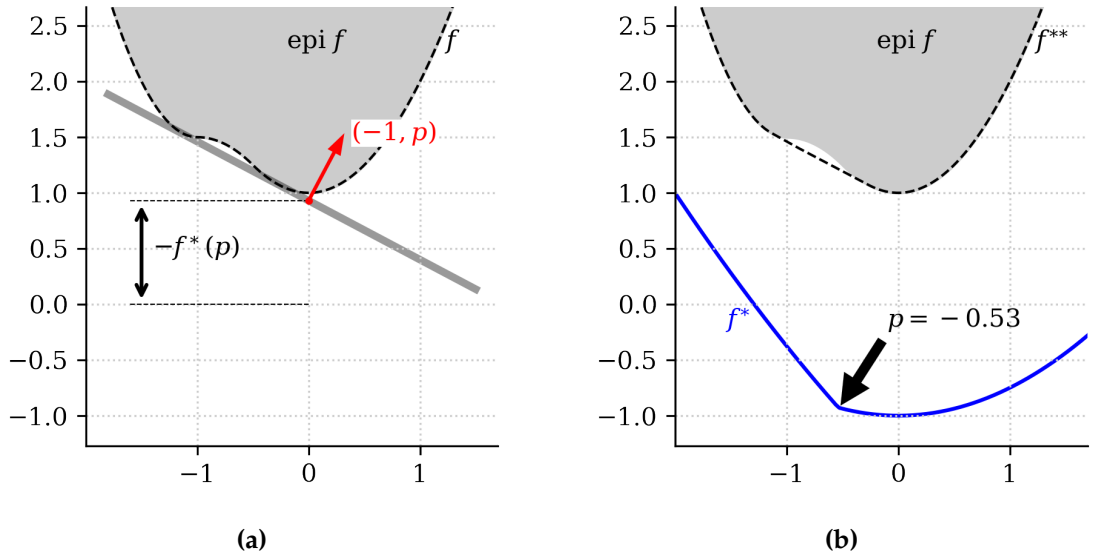


Figure 2.3. Examples of a non-convex function f and the corresponding convex conjugate and biconjugate functions. (a) The function f (dashed line) and the geometry of the corresponding convex conjugate f^* for $p = 0.53$. (b) The convex conjugate f^* (blue line) and the corresponding biconjugate f^{**} (dashed line).

$\varphi\colon \mathbb{R}^n \times \mathbb{R}^m \to \overline{\mathbb{R}}$ and define $v(u) = \inf_{x\in\mathbb{R}^n} \varphi(x,u)$. Using the definition in Eq. (2.9) yields the convex conjugate

$$\begin{aligned} v^*(p) &= \sup_{u\in\mathbb{R}^m} \sup_{x\in\mathbb{R}^n} \langle u,p\rangle - \varphi(x,u) \\ &= \sup_{\substack{x\in\mathbb{R}^n \\ u\in\mathbb{R}^m}} \langle 0,x\rangle + \langle u,p\rangle - \varphi(x,u) = \varphi^*(0,p)\,. \end{aligned} \tag{2.10}$$

For any auxiliary function φ satisfying $\varphi(x,0) = f(x)$, the *primal problem*

$$v(0) = \inf_{x\in\mathbb{R}^n} \varphi(x,0) = \inf_{x\in\mathbb{R}^n} f(x) \tag{2.11a}$$

is obtained. As a function of $u \in \mathbb{R}^m$, the auxiliary function $\varphi(x,u)$ perturbs the problem and is thus called *perturbation function*. The vector $x \in \mathbb{R}^n$ in Eq. (2.11a) denotes the *primal variable*.

The definition of the biconjugate and Eq. (2.10) directly yield

$$v^{**}(0) = \sup_{p\in\mathbb{R}^m} -\varphi^*(0,p), \tag{2.11b}$$

which is known as the *dual problem* of the primal problem in Eq. (2.11a). The vector $p \in \mathbb{R}^m$ in Eq. (2.11b) denotes the *dual variable*.

The relation $v \geq v^{**}$, which is due to Fenchel's inequality, is known as *weak duality*. The difference $v(0) - v^{**}(0)$ is the *duality gap* of the problem. In the special case $v^{**} = v$, it is said that *strong duality* holds (the duality gap is zero).

Lagrange duality

To obtain Lagrange duality, we consider a primal problem with m constraints,

$$\begin{aligned} &\text{minimize} && f_0(x) \\ &\text{subject to} && f_i(x) \leq 0 \text{ for all } i = 1, \ldots, k, \\ &&& f_i(x) = 0 \text{ for all } i = k+1, \ldots, m. \end{aligned} \tag{2.12}$$

and suppose that $\operatorname{dom} f_0 = \mathbb{R}^n$ and that the feasible set of the problem in Eq. (2.12) is non-empty.

Using the indicator function $\delta_K \colon \mathbb{R}^n \to \overline{\mathbb{R}}$ as defined in Eq. (2.3), we can implicitly encode the constraints from Eq. (2.12) into the objective function and obtain

$$\inf_{x \in C} f_0(x) = \inf_{F(x) \in K} f_0(x) = \inf_{x \in \mathbb{R}^n} f_0(x) + \delta_K(F(x)), \tag{2.13}$$

the form in Eq. (2.11a), where $K = \mathbb{R}^k_- \times \{0\}^{m-k}$ and $F = (f_1, \ldots, f_m)$.

Let $\varphi(x, u) = f_0(x) + \delta_K(F(x) + u)$ be the perturbation function. By calculating its convex conjugate φ^*, we obtain the dual problem

$$v^{**}(0) = \sup_{p \in \mathbb{R}^m} \inf_{x \in \mathbb{R}^n} L(x, p) - \delta_K^*(p), \qquad L(x, p) = f_0(x) + \langle F(x), p \rangle \tag{2.14}$$

due to Eq. (2.11b), where $L(x, p)$ is called the *Lagrangian*.

From the definition of the indicator function in Eq. (2.3) and the definition of the convex conjugate in Eq. (2.9), it can be shown that the conjugated indicator function δ_K^* with $K = \mathbb{R}^k_- \times \{0\}^{m-k}$ resolves to the indicator function δ_{K^*} with $K^* = \mathbb{R}^k_+ \times \mathbb{R}^{m-k}$. Thus, the dual problem reads

$$v^{**}(0) = \sup_{p \in \mathbb{R}^m} \inf_{x \in \mathbb{R}^n} L(x, p) - \delta_{K^*}(p) = \sup_{p \in \mathbb{R}^k_+ \times \mathbb{R}^{m-k}} \inf_{x \in \mathbb{R}^n} L(x, p) \tag{2.15a}$$

and the Lagrangian from Eq. (2.14) takes the form

$$L(x, p) = f_0(x) + \langle F(x), p \rangle = f_0(x) + \sum_{i=1}^{m} p_i \cdot f_i(x). \tag{2.15b}$$

The dual variables $p_1, \ldots, p_m$ are the *Lagrange multipliers*. For $i = 1, \ldots, k$, the Lagrange multipliers p_i are associated with the primal inequality constraints and constrained by $p_i \geq 0$. The remaining Lagrange multipliers $p_{k+1}, \ldots, p_m$ are associated with the primal equality constraints and unconstrained.

2.2.4 KKT conditions

We assume for this section that the primal and dual optimal solutions are attained, i.e. there is a primal-dual solution $(\hat{x}, \hat{p}) \in \mathbb{R}^n \times \mathbb{R}^k_+ \times \mathbb{R}^{m-k}$ so that $v^{**}(0) = L(\hat{x}, \hat{p})$, and that the functions $f_0, \ldots, f_m$ are twice differentiable.

Let $\hat{x}$ and $\hat{p}$ be any primal and dual optimal solutions. Since $\hat{x}$ solves $\inf_{x \in \mathbb{R}^n} L(x, \hat{p})$ by supposition, it follows immediately that $\nabla_x L(\hat{x}, \hat{p}) = 0$, i.e.

$$\nabla_x L(\hat{x}, \hat{p}) = \nabla f_0(\hat{x}) + \sum_{i=1}^{m} \hat{p}_i \cdot \nabla f_i(\hat{x}) = 0. \tag{2.16}$$

This condition is one from a set of conditions, which generalize the *necessary* optimality conditions for unconstrained problems to the constrained case (e.g., [15]). Another less obvious condition is obtained by considering the primal problem $v(0) = \inf_{x \in \mathbb{R}^n} f(x)$ in Eq. (2.13) with the duality gap $\Delta v = v(0) - v^{**}(0)$. Then,

$$v(0) = f_0(\hat{x}) = v^{**}(0) + \Delta v = L(\hat{x}, \hat{p}) + \Delta v \tag{2.17a}$$

$$= f_0(\hat{x}) + \sum_{i=1}^{m} \hat{p}_i \cdot f_i(\hat{x}) + \Delta v, \tag{2.17b}$$

and since $f_i(\hat{x}) = 0$ for all $i = k+1, \ldots, m$,

$$0 = \sum_{i=1}^{k} \hat{p}_i \cdot f_i(\hat{x}) + \Delta v. \tag{2.17c}$$

Thus, if strong duality holds (i.e. $\Delta v = 0$), then the equation $0 = \sum_{i=1}^{k} \hat{p}_i \cdot f_i(\hat{x})$ is obtained. Since $f_i(\hat{x}) \leq 0$ and $\hat{p}_i \geq 0$ for all $i = 1, \ldots, k$ due to feasibility of $\hat{x}$ and $\hat{p}$, it immediately follows that $\hat{p}_i \cdot f_i(\hat{x}) = 0$ for all $i = 1, \ldots, k$, which is called *complementary slackness*.

The necessary optimality conditions for problems with strong duality, better known as the *Karush-Kuhn-Tucker (KKT) conditions*, are summarized as follows:

Feasibility condition. The solution $\hat{x}, \hat{p}$ is primal and dual feasible, i.e. $f_1(\hat{x}), \ldots, f_k(\hat{x}) \leq 0$, $f_{k+1}(\hat{x}), \ldots, f_m(\hat{x}) = 0$, and $\hat{p}_1, \ldots, \hat{p}_k \geq 0$.

Stationary condition. The solution $\hat{x}, \hat{p}$ is a stationary point of the Lagrangian, i.e. the solution satisfies Eq. (2.16).

Complementary slackness. The condition $\hat{p}_i \cdot f_i(\hat{x}) = 0$ holds for all inequality constraints, indexed by $i = 1, \ldots, k$.

These conditions describe a solution $\hat{x}, \hat{p}$ if it is optimal.

If the functions $f_0, \ldots, f_k$ are convex, the functions $f_{k+1}, \ldots, f_m$ are affine, and $\hat{p}$ is dual feasible, then the Lagrangian $L(x, \hat{p})$ in Eq. (2.15b) is a convex function in

x (see the rules for weighted and non-negative weighted sums in Section 2.2.2). If, in addition, the *stationary condition* holds, then the solution $\hat{x}, \hat{p}$ minimizes the Lagrangian (due to the first-order condition in Section 2.2.2), i.e.

$$v^{**}(0) = \inf_{x \in \mathbb{R}^n} L(x, \hat{p}) = L(\hat{x}, \hat{p}) = f_0(\hat{x}) + \sum_{i=1}^{k} \hat{p}_i \cdot f_i(\hat{x}). \tag{2.18}$$

Finally, if *complementary slackness* holds too, then the above equation boils down to $v^{**}(0) = f_0(\hat{x})$. This means that $\hat{x}$ and $\hat{p}$ have zero duality gap, i.e. the solution $\hat{x}, \hat{p}$ is primal and dual optimal. Thus, for optimization problems with convex objective functions, convex inequality constraints, and affine equality constraints, the KKT conditions are not only necessary, but also *sufficient* optimality conditions (e.g., [15]).

2.2.5 Convex optimization

Due to the sufficiency of the KKT conditions described above, optimization of problems with convex objective functions, convex inequality constraints, and affine equality constraints boils down to solving the KKT conditions.

Below, we consider the primal-dual space $\mathbb{R}^n \times \mathbb{R}^m$. The *primal feasible region* $\{x \in \mathbb{R}^n | F(x) \in K\}$ is the intersection of the solution sets of the inequalities $f_1, \ldots, f_k$ and the equalities $f_{k+1}, \ldots, f_m$. The *dual feasible region* is $\mathbb{R}^k_+ \times \mathbb{R}^{m-k}$. Overall, $\{x \in \mathbb{R}^n | F(x) \in K\} \times \mathbb{R}^k_+ \times \mathbb{R}^{m-k}$ is the *feasible region* of the primal-dual space.

Direct solution of the KKT conditions

The KKT conditions pose a system of non-linear equations (except for the inequalities $f_1(\hat{x}), \ldots, f_k(\hat{x}) \leq 0$ and $\hat{p}_1, \ldots, \hat{p}_k \geq 0$). This system can be solved iteratively using Newton's method, which is based on local linearization of the non-linear system (first-order Taylor approximation).

Let $y = (x, p)$ be the current iterate and $y + \Delta y$ the next iterate, so that $\Delta y = (\Delta x, \Delta p)$ corresponds to the primal-dual Newton step. Then, for example, the local linearization of complementary slackness is

$$p_i \cdot f_i(x) + \Delta p_i \cdot f_i(x) + p_i \cdot \nabla f_i(x) \cdot \Delta x = 0. \tag{2.19}$$

Once the iterate reaches the boundary of the dual feasible region ($p_i = 0$), Eq. (2.19) yields $\Delta p_i \cdot f_i(x) = 0$, i.e. $\Delta p_i = 0$ if $f_i(x) < 0$. For the boundary of the primal feasible region ($f_i(x) = 0$), Eq. (2.19) yields $\nabla f_i(x) \cdot \Delta x = 0$, i.e. $\nabla f_i(x) \perp \Delta x$. Thus, the iterate sticks to the boundary of the feasible region, once it is reached, potentially precluding the global convergence of the method (see, e.g., [24]).

Interior point methods

To avoid sticking to the boundary of the feasible region and improve global convergence, the KKT solutions are not solved directly. Instead, the standard approach is to consider a sequence of *perturbed* conditions, which *converge* to the KKT conditions (e.g., [15]). To this end, complementary slackness is replaced by

$$-\hat{p}_i \cdot f_i(\hat{x}) = 1/t, \tag{2.20}$$

where the parameter $t > 0$ is incrementally increased, successively decreasing the perturbation and eventually converging to complementary slackness (for $t \to \infty$). Since both $f_i(\hat{x}) = 0$ and $\hat{p}_i = 0$ contradict Eq. (2.20), the iterate approaches the primal-dual solution $\hat{y}$ via the *interior* of the region which corresponds to the solution set of the primal and dual inequality constraints. Such approaches are thus referred to as *interior point methods*.

Replacing complementary slackness by the perturbed condition in Eq. (2.20) yields the *modified KKT conditions* $r_t(x, p) = 0$, where

$$r_t(x,p) = \left[\begin{array}{c|l} \nabla_x^\top L(x,p) & \text{Stationary condition (cf. Eq. (2.16))} \\ \hline -p_1 \cdot f_1(x) - 1/t & \\ \vdots & \text{Perturbed complementary slackness} \\ -p_k \cdot f_k(x) - 1/t & \\ \hline f_{k+1}(x) & \\ \vdots & \text{Equality constraints } r_t^{\text{eq}}(x,p) \\ f_m(x) & \end{array}\right] \tag{2.21}$$

is the residuals vector of the conditions, consisting of three blocks, and $r_t^{\text{eq}}(x,p) = (f_{k+1}(x), \ldots, f_m(x))$ is the bottom block (corresponding to the equality constraints).

In Eq. (2.21), only the equality constraints are regarded. The non-linear system $r_t(x,p) = 0$ is solved subject to $F(x) \in \mathbb{R}^k_-$ and $p \in \mathbb{R}^k_+ \times \mathbb{R}^{m-k}$ using a Newton-based scheme, which explicitly handles the inequality constraints $F(x) \in \mathbb{R}^k_-$ and $p \in \mathbb{R}^k_+ \times \mathbb{R}^{m-k}$. Given the current iterate $y = (x,p)$, the Newton step Δy is obtained by solving $r_t(y + \Delta y) = 0$ for Δy using the first-order Taylor approximation

$$r_t(y + \Delta y) \approx r_t(y) + \nabla r_t(y) \cdot \Delta y, \tag{2.22}$$

where ∇r_t is the Jacobian of Eq. (2.21). Supposing that ∇r_t is strictly positive definite and thus invertible, we obtain the *primal-dual interior point approach* for constrained convex optimization (e.g., [15]), which comprises three main steps in each iteration:

1. The Newton step $\Delta y = -\nabla r_t(y)^{-1} \cdot r_t(y)$ corresponding to the non-linear system of equations $r_t(y + \Delta y) = 0$ is obtained from Eq. (2.22).

2. The largest admissible step length $\lambda \in (0, 1)$ is determined using line search, so that *(i)* feasibility is retained with respect to the primal and dual inequalities for the next iterate $y + \lambda \cdot \Delta y$, and *(ii)* the residuals $\|r_t(y + \lambda \cdot \Delta y)\|_2$ are reduced in comparison to $\|r_t(y)\|_2$.

3. The current iterate $y \leftarrow y + \lambda \cdot \Delta y$ is updated.

Using an initialization of the iterate y which is feasible with respect to the primal and dual inequality constraints, the above scheme retains this feasibility. Note, however, that a single iteration does generally not yield an iterate which is feasible with respect to the equality constraints.

Thus, solving Eq. (2.17c) for the duality gap $\Delta\nu$ yields the *surrogate duality gap* $\Delta\nu(y) = -\sum_{i=1}^{k} p_i \cdot f_i(x)$, which is an approximation of the real duality gap due to $r_t^{\text{eq}}(y) \neq 0$. The iterations terminate when $\Delta\nu(y)$ is small *and* converges to the real duality gap (i.e. $\left\|r_t^{\text{eq}}(y)\right\|_2$ is sufficiently small too). The parameter t is updated at the beginning of each iteration based on the duality gap of the current iterate.

The primal-dual interior point method for constrained convex optimization is a second-order method, since second-order information of the primal problem is exploited via the gradient of the Lagrangian in Eq. (2.21) and the corresponding Jacobian ∇r_t in the Newton step. The method is closely related to barrier methods, which solve the modified KKT conditions using *multiple* Newton steps per iteration. For such methods, the overall number of steps grows linearly with the number k of inequality constraints, and logarithmically with the inverse of the precision (thresholds of the termination criteria). The primal-dual interior point method exhibits faster convergence in practice than barrier methods. The overall run time is dominated by the computation of the inverse of the Jacobian, which is cubic in $n + m$ for both barrier and primal-dual interior point methods.

Unconstrained convex optimization

Unconstrained convex optimization is a special case of constrained convex optimization (using $k = m = 0$ constraints). In this case, primal-dual interior point methods cannot be used directly, since neither the surrogate duality gap nor the residuals of the equality constraints can be used as a termination criterion. To cope with this, an unconstrained convex optimization problem $\inf_{x \in \mathbb{R}^n} f_0(x)$ can be transformed into the constrained form in Eq. (2.12). To this end, note that any (x, μ) with $x \in \mathbb{R}^n$ and $\mu \in \mathbb{R}$ which satisfies $\mu \geq f_0(x)$ is a point of the epigraph of f_0 (cf. Eq. (2.5)). Minimization of the scalar value μ subject to the constraint $\mu \geq f_0(x)$ hence also minimizes $f_0(x)$. This motivates the *epigraph* form

$$\begin{aligned} &\text{minimize} && \mu \\ &\text{subject to} && f_0(x) - \mu \leq 0 \end{aligned} \tag{2.23}$$

of $\inf_{x \in \mathbb{R}^n} f_0(x)$, where the feasible set of the constrained problem in Eq. (2.23) corresponds to the epigraph of f_0 (e.g., [15]). The inequality constraint in Eq. (2.23) is convex since it is a sum of the convex functions f_0 and $-\mu$ (cf. Section 2.2.2). The constrained convex problem can be solved using primal-dual interior point methods for constrained convex optimization with respect to $(x, \mu) \in \mathbb{R}^{n+1}$.

Linear programming

Another special case of constrained convex optimization is *linear programming* (e.g., [15]). If the functions $f_0, \ldots, f_m$ are affine, the problem is called a *linear program (LP)*. In the form of Eq. (2.12) an LP can be written

$$\begin{aligned} &\text{minimize} && d + \langle c, x \rangle \\ &\text{subject to} && G \cdot x \leq h, \\ & && A \cdot x = b, \end{aligned} \tag{2.24}$$

where $d \in \mathbb{R}$ and $c \in \mathbb{R}^n$ are the *weights* of the LP. The constraints of the LP are specified using the matrices $G \in \mathbb{R}^{k \times n}$ and $A \in \mathbb{R}^{m-k \times n}$ as well as the vectors $h \in \mathbb{R}^k$ and $b \in \mathbb{R}^{m-k}$ of corresponding dimensions.

Definition 2.1 (Polytope associated with an LP). The feasible set P associated with an LP is the solution set of the constraints in Eq. (2.24). Due to the affinity of the constraints, the feasible set P of an LP is a closed *polytope*.

We assume that the objective function f_0 is linear (i.e. $d = 0$) without loss of generality due to $\inf_{x \in P} d + \langle c, x \rangle = d + \inf_{x \in P} \langle c, x \rangle$. Since a polytope is a convex set and any linear objective function also is convex, the primal-dual interior point-based approach described above can be used to solve an LP (dedicated LP solvers exploit the linear structure, e.g., [25]).

For further considerations, we note that the dual problem of an LP takes a simple form (e.g., [16, 26]). If there are no primal equality constraints ($m = k$), then Eq. (2.15) yields the dual problem

$$v^{**}(0) = \sup_{p \in \mathbb{R}^k_+} \inf_{x \in \mathbb{R}^n} \langle c, x \rangle + \langle G \cdot x, p \rangle - \langle h, p \rangle \tag{2.25a}$$

$$= \sup_{p \in \mathbb{R}^k_+} -\langle h, p \rangle + \inf_{x \in \mathbb{R}^n} \left\langle c + G^\top \cdot p, x \right\rangle \tag{2.25b}$$

$$= \sup_{p \in \mathbb{R}^k_+} -\langle h, p \rangle - \begin{cases} 0 & \text{if } G^\top \cdot p = -c \\ \infty & \text{otherwise} \end{cases} \tag{2.25c}$$

of the LP in Eq. (2.24). The dual problem in Eq. (2.25c) can be written as an LP too,

$$\begin{aligned} \text{maximize} \quad & -\langle h, p \rangle \\ \text{subject to} \quad & p \geq 0, \\ & G^\top \cdot p = -c, \end{aligned} \tag{2.26}$$

since the last term in Eq. (2.25c) corresponds to the definition of the indicator function in Eq. (2.3) for the solution set of the equality constraints in Eq. (2.26).

2.2.6 Combinatorial optimization

We consider a primal problem of the form in Eq. (2.24), which has no equality constraints but exhibits the additional integrality constraint $x \in \{0, 1\}^n$,

$$\begin{aligned} \text{minimize} \quad & \langle c, x \rangle \\ \text{subject to} \quad & G \cdot x \leq h, \\ & x \in \{0, 1\}^n \end{aligned} \tag{2.27}$$

and is denoted an *integer linear program (ILP)*. Replacing the integrality constraint by the inequality constraint $0 \leq x \leq 1$ yields the *LP relaxation* of the ILP in Eq. (2.27).

Due to the integrality constraint, the objective function of the ILP corresponds to a sum of a *subset* of the elements of the vector c, which naturally gives rise to the interpretation of Eq. (2.27) as a *combinatorial* optimization problem. However, the feasible set of an ILP is not convex due to the integrality constraint. Thus, an ILP does not exhibit the computationally favorable properties which could be exploited by convex programming. Many famous combinatorial problems can be expressed in the form of the ILP in Eq. (2.27), including *min-weight set-cover* and *max-weight set-packing* (see examples below). Most combinatorial problems are **NP**-hard, thus, in general, determining the optimal solution is not tractable (cf. Section 2.2.1).

Set-cover and set-packing

Two closely related examples of **NP**-hard optimization problems in the form of the ILP in Eq. (2.27) are given below (e.g., [17, 26]).

For simplicity, we consider a universe $U = [m]$ of integers, which, however, can be used as an index set to identify *any* kind of elements (e.g., image regions). Also, let $\mathscr{S}$ be a family of sets $X_1, \ldots, X_n \subseteq U$, and let $c \in \mathbb{R}^n_+$ be a vector of non-negative *weights*, where the k-th component c_k corresponds to the weight associated with the k-th set X_k. To encode the interrelations of the universe U and family $\mathscr{S}$ in matrix representation, we define the $m \times n$ matrix G', where $G'_{ik} = [i \in X_k]$.

Definition 2.2 (Set cover). A *set cover* is a subset $\mathscr{X} \subseteq \mathscr{S}$, which covers each element of the universe U. A *min-weight set-cover* is a set cover, which is minimal

with respect to the total weights $\sum_{k\in[n]} c_k \cdot [X_k \in \mathscr{X}]$ associated with the included sets. The ILP

$$\begin{aligned} &\text{minimize} && \langle c, x\rangle \\ &\text{subject to} && G' \cdot x \geq 1, \\ & && x \in \{0,1\}^n \end{aligned} \tag{2.28}$$

corresponds to the problem of determining a min-weight set-cover.

Definition 2.3 (Set packing). A *set packing* is a subset $\mathscr{X} \subseteq \mathscr{S}$ which covers no element of the universe U more than once (i.e. the sets in $\mathscr{X}$ are disjoint). A *max-weight set-packing* is a set packing, which is maximal with respect to the total weights $\sum_{k\in[n]} c_k \cdot [X_k \in \mathscr{X}]$ associated with the included sets. The ILP

$$\begin{aligned} &\text{maximize} && \langle c, x\rangle \\ &\text{subject to} && G' \cdot x \leq 1, \\ & && x \in \{0,1\}^n \end{aligned} \tag{2.29}$$

corresponds to the problem of determining a max-weight set-packing.

Any vector $x \in \{0,1\}^n$ directly yields a family $\mathscr{X} = \{X_k \in \mathscr{S} | k \in [n], x_k = 1\}$ and vice versa. If x is a feasible solution of the ILPs in Eq. (2.28) and Eq. (2.29), then $\mathscr{X}$ is the corresponding set cover or set packing, respectively.

Approximation using LP relaxations and dual problems

To cope with the **NP**-hardness of combinatorial problems, methods have been developed which exploit the specific problem structure and determine approximative solutions, which are *guaranteed* to be close to global optimality.

To quantify the approximation accuracy, the value APX of the approximative solution must be compared to the value OPT of the optimal solution, which is unknown. However, solving the LP relaxation of an ILP yields a *fractional* solution and a corresponding lower bound $\mathrm{OPT_f}$ of the optimal value OPT of the ILP,

$$\inf_{\substack{x\in\mathbb{R}^n \\ G\cdot x\leq h}} \langle c, x\rangle = \mathrm{OPT_f} \leq \inf_{\substack{x\in\{0,1\}^n \\ G\cdot x\leq h}} \langle c, x\rangle = \mathrm{OPT} \leq \mathrm{APX}, \tag{2.30a}$$

which is due to the feasible set of the ILP being a subset of the feasible set C of the LP relaxation (e.g., [26]). Let P_P and P_D denote the polytopes associated with the LP relaxation of the primal problem and its dual (cf. Definition 2.1). Then, taking the dual problem in Eq. (2.26) of the LP into consideration, we obtain the inequality

$$-\langle h, p\rangle \leq \mathrm{OPT_f} \leq \mathrm{OPT} \leq \langle c, x\rangle = \mathrm{APX}, \tag{2.30b}$$

which holds for all $x \in P_\mathrm{P}$ and $p \in P_\mathrm{D}$ satisfying $x \in \{0,1\}^n$. The *approximation ratio* APX/OPT is a measure of the approximation accuracy (lower is better). An approximation ratio of 1 means that the optimal solution is attained.

In *a posteriori* analysis, Eq. (2.30a) is used to determine an *upper bound* of the approximation ratio, i.e. $\mathrm{APX/OPT} \leq \mathrm{APX/OPT_f}$, by solving the corresponding LP relaxation to obtain $\mathrm{OPT_f}$. This means that the approximation accuracy of a given approximative solution APX is $\mathrm{APX/OPT_f}$ *or better*. On the other hand, Eq. (2.30b) is useful to *a priori* establish an upper bound of the approximation ratio, i.e. $\mathrm{APX/OPT} \leq \langle c, x\rangle / -\langle h, p\rangle$. This is the worst-case approximation ratio, a so called *approximation guarantee*. A prominent example is given below.

Approximative solution of min-weight set-cover

It can be seen from Eq. (2.26), that the dual problem of the LP relaxation of the problem in Eq. (2.28) is to maximize $\langle 1, p\rangle$ with respect to $p \in \mathbb{R}^m_+$ subject to the system of equalities $G'^\top \cdot p = c$ (due to $G = -G'$). Since *(i)* the objective is to maximize the components of the vector p, and since *(ii)* both p and the matrix G' only have non-negative entries, the equality constraints can be written as inequalities. This leads to

$$\begin{aligned} \text{maximize} \quad & \langle 1, p\rangle \\ \text{subject to} \quad & p \geq 0, \\ & G'^\top \cdot p \leq c \end{aligned} \tag{2.31}$$

as the dual problem of the LP-relaxation of the *min-weight set-cover* problem. Notably, up to an anisotropic scaling, the feasible set of the LP in Eq. (2.31) corresponds to the polytope associated with the LP relaxation of the *max-weight set-packing* problem defined in Eq. (2.29).

In the dual problem in Eq. (2.31), the components of the vector p are associated with the elements $1, \ldots, m$ of the universe. Due to the structure of the matrix G', the dual inequality constraint $G'^\top \cdot p \leq c$ means that

$$\sum_{i \in X_k} p_i \leq c_k \tag{2.32}$$

for all sets enumerated by $k = 1, \ldots, n$, i.e. the sum of components of p corresponding to a single set cannot exceed the weight of that set.

The *greedy* algorithm [27] for approximation of min-weight set-cover iteratively covers the elements of the universe by adding a *specific* set of the family $\mathscr{S}$ to the cover $\mathscr{X}$ in each iteration (initially $\mathscr{X} = \emptyset$) and terminates when all elements are covered (i.e. x is a feasible solution and thus $\bigcup \mathscr{X} = U$). Let $V^{(t)} \subseteq U$ be the set of already covered elements in iteration t and $V^{(1)} = \emptyset$. The set $X_{\hat{k}(t)}$ which is added to the cover $\mathscr{X}$ in iteration t, is determined by $\hat{k}(t) = \arg\min_{k \in [n]} c_k / r_k^{(t)}$, where

$r_k^{(t)} = \#\left(X_k \setminus V^{(t)}\right)$ is the number of newly covered elements, and thus

$$c_{\hat{k}(t)}/r_{\hat{k}(t)}^{(t)} \leq c_k/r_k^{(t)} \tag{2.33}$$

for all $k \in [n]$. Consider the set X_k for a fixed but arbitrary $k \in [n]$ and let i_1, $\ldots$, $i_{\#X_k}$ be the order in which the elements of X_k are covered by the algorithm (the order is arbitrary for elements covered within the same iteration). When an arbitrary element i_j is covered for the first time, there are currently $\#X_k + 1 - j$ uncovered elements in X_k. This occurs in iteration t, and the dual variable p is updated so that the weight $c_{\hat{k}(t)}$ of the set added in this iteration is distributed among the $r_{\hat{k}(t)}^{(t)}$ newly covered elements. This yields the assignment

$$p_{i_j} \leftarrow \lambda \cdot c_{\hat{k}(t)}/r_{\hat{k}(t)}^{(t)} \tag{2.34}$$

when the element i_j is covered, where $\lambda > 0$ is a fixed factor. Notably, $p_{i_j} \leq \lambda \cdot c_k/r_k^{(t)}$ due to Eq. (2.33), and $r_k^{(t)} \geq \#X_k + 1 - j$, since i_j might be not the first element being covered in iteration t (usually multiple elements are covered within a single iteration). Considering the lhs of Eq. (2.32) thus yields

$$\sum_{i \in X_k} p_i \leq \lambda \cdot \sum_{j \in [\#X_k]} c_k/(\#X_k + 1 - j) = \lambda \cdot c_k \cdot H_{\#X_k}, \tag{2.35a}$$

where $H_{\#X_k} = 1 + 1/2 + \cdots + 1/\#X_k$ is the harmonic number, and

$$\sum_{i \in X_k} p_i \leq \lambda \cdot c_k \cdot H_m, \tag{2.35b}$$

since $\#X_k \leq m$. Comparing Eq. (2.35b) to Eq. (2.32) shows that the dual variable p is feasible if $\lambda = 1/H_m$. The primal solution x is obviously feasible due to the termination criterion. Due to primal and dual feasibility of x and p, Eq. (2.30b) can be used to deduce the approximation guarantee. Note that for any set added to the cover $\mathscr{X}$, the algorithm also increases the dual variable p by the weight of that set scaled by the factor $\lambda = 1/H_m$, due to Eq. (2.34). Thus, primal and dual solutions are coupled by $\langle 1, p \rangle = \langle c, x \rangle / H_m$, which yields the approximation guarantee

$$\frac{\mathrm{APX}}{\mathrm{OPT}} \leq \frac{\langle c, x \rangle}{\langle c, x \rangle / H_m} = H_m. \tag{2.36}$$

This means that the greedy algorithm for approximative solution of min-weight set-cover yields a solution, which is *at most* of a factor H_m worse than the optimal solution (the decay is less than logarithmic with respect to m). We refer to [26] for further details.

2.3 Overview of existing approaches for cell segmentation

In this section, we give an overview of previous work on cell segmentation. Besides standard segmentation approaches (Section 2.3.1), one can generally distinguish between model-based (Section 2.3.2) and learning-based approaches (Section 2.3.3). Model-based approaches perform inference by energy minimization as opposed to learning-based approaches, where inference typically amounts to forward computations using a previously optimized model [28].

2.3.1 Standard approaches

Standard approaches for cell segmentation often employ thresholding methods (e.g., [29, 30, 31, 32]). Wu et al. [29] proposed two-stage thresholding based on local variation of the image intensities. In [30, 31, 32], adaptive thresholding based on the histogram of the image intensities [33] was used. Singh et al. [30] determined individual objects by connected component analysis of the binarized image in order to study different cellular phenotypes in fluorescence microscopy images. To better cope with closely clustered objects, Singh et al. [31] proposed seeded region growing to split falsely merged objects in the binarized image, which was supplemented by variational level sets for final refinement. Accurate initialization of the region growing step is required to prevent falsely merged and split objects. He et al. [32] determined concave points on the boundary of the connected components of the binarized image. Then, modified DBSCAN [34, 35] was used for clustering of the concave points to split falsely merged objects. This has the advantage that initialization is not required, however, it is prone to cell nuclei with strongly non-elliptical shapes. Other standard approaches employed the watershed transform or morphological analysis (e.g., [36, 37, 38]). Wählby et al. [36] proposed a two-level watershed transform to obtain an initial segmentation of fluorescence microscopy images. The result was then refined by recognition of falsely merged or split objects, according to 19 different appearance features. Cheng et al. [37] described a watershed-based approach for cell cluster splitting based on the distance transform of segmentation masks. Plissiti et al. [38] used morphological reconstruction for detection of cell nuclei and morphological gradients for subsequent detection of the nuclei boundaries. Generally, standard segmentation approaches are sensitive to texture, prone to image noise, and intensity inhomogeneities.

2.3.2 Model-based approaches

A broad class of cell segmentation methods are *model-based approaches*. In these approaches, models of an object or an image are gradually configured (i.e. adapted) so that the model fits the image data. Model configurations are often realized via parameterization or representation in a variational framework (see below). Different model configurations are scored by an *energy* value, which is small for

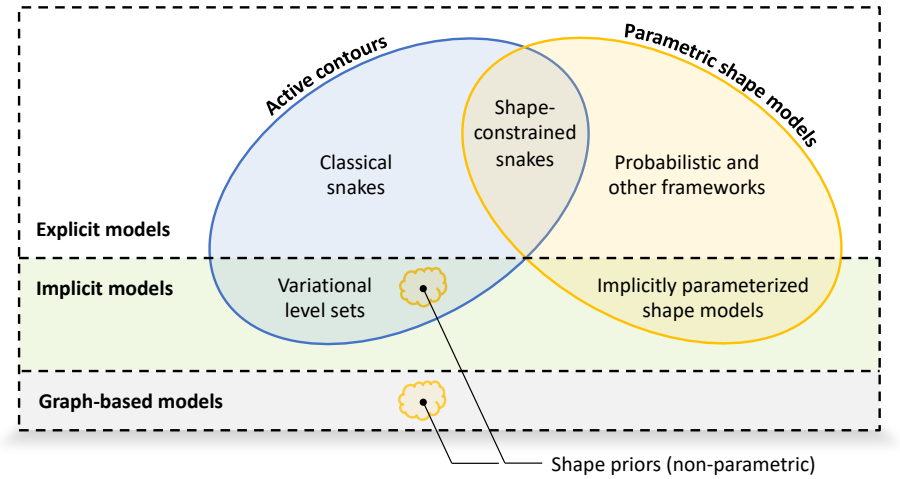


Figure 2.4. Diagram of the models used in previous model-based approaches.

configurations which fit the image data, and large otherwise. Model fitting, i.e. *inference* of the optimal model configuration from the image data, corresponds to *energy minimization* with respect to the configuration of the model [28]. Globally optimal configurations yield the models which best fit the image data (depending on, e.g., the parameterization or the particular quantification of the energy values, which differ heavily among the methods).

However, since computationally tractable optimization requires exploitation of computational properties (e.g., convexity, cf. Section 2.2.1), the majority of previous model-based cell segmentation methods only used *local optimization*. Such methods aim to solve a relaxed variant of the optimality condition in Eq. (2.2), which only holds locally (cf. Section 2.2). If the initialization of the local optimization method is close to the global solution, then the obtained locally optimal solution *might* also be globally optimal (this depends on the optimization method). For this reason, accurate initialization of local optimization-based segmentation methods is crucial.

An overview of different types of model-based approaches is given in Figure 2.4. Two major classes of model-based approaches are active contours and parametric shape models. One can also differentiate between explicit models (e.g., snakes and explicitly parameterized shape models), implicit models (e.g., variational level sets and implicitly parameterized shape models), and graph-based models. Below, we describe parametric active contours (snakes), non-parametric active contours (variational level sets), parametric shape models, and graph-based models.

Snakes and parametric active contours

Kass et al. [39] introduced *snakes*, which are explicitly parameterized curves associated with an energy function based on the image gradient. Minimization of the energy function drives the snake towards image points corresponding to large image gradient magnitude (e.g., lines and edges), while maintaining a reasonable

smoothness and length of the curve due to regularization. Snakes are also called *parametric active contours* or *parametric deformable models*.

Snake-based methods have been successfully used for cell segmentation (e.g., [40, 41, 42, 43, 44]). Zimmer and Olivo-Marin [40] proposed an energy function which incorporates multiple snakes simultaneously. They also employed an object interaction model (penalization of object overlap), which allows for joint cell segmentation and cluster splitting. To better cope with partial object overlap and image distortions like strong image noise, approaches were developed which incorporate *shape information*. For example, Thevenaz and Unser [41] proposed a parameterization for circular snakes and an energy function which incorporates the image intensities. This method was used for vessel segmentation and cell counting. In [42], they described an extension for cell segmentation based on elliptical shapes. Delgado-Gonzalo et al. [43] introduced a spline-based snake parameterization. This parameterization represents elliptical shapes, but allows for deformations, and copes well with non-elliptical cell nuclei. In [44], they extended the method for segmentation of biomedical 3-D data. However, no snake-based method mentioned above exploited shape information for joint cell segmentation and cluster splitting (the methods [41, 42, 43, 44] did not use the energy function incorporating multiple snakes previously proposed in [40]). Only local optimization methods were used for energy minimization. Initial placement of the snakes is thus crucial since globally optimal solutions were not obtained.

Non-parametric active contours

Segmentation approaches based on *implicit* models were developed concurrently to the explicitly parameterized snakes. In the variational level set framework [45, 46], object and model contours are implicitly represented as level sets of non-parametric functions, the so-called *level set functions* (e.g., a function corresponding to the signed distance of an image point to the contour of the model). Variational level sets are thus also called *non-parametric active contours* or *non-parametric deformable models*. Model fitting is performed by minimization of a suitable energy functional in function space with respect to the level set function, like the predominant Chan and Vese [47] functional. Advantages of variational level set methods over snakes are that the results are obtained independently of the parameterization and that topological changes are handled naturally.

Variational level set methods for cell segmentation (e.g., [48, 49, 50, 51, 52]) have been popular alternatives of snake-based methods. De Solorzano et al. [48] used a variational level set approach for segmentation of cell nuclei in low-contrast microscopy data. Analogous to the snake energy for joint cell segmentation and cluster splitting [40], Dufour et al. [49] proposed an extension of the variational level set framework using a similar object interaction model (both works were published in the same year). Li et al. [50] introduced an energy functional which

is robust to intensity inhomogeneities and strong image noise. Since only local optimization methods were used in [48, 49, 50, 52], the initialization of the level set functions was crucial. To overcome this, Bergeest and Rohr [51] proposed a globally optimal approach for cell segmentation based on *convex* energy minimization [53, 54] and the energy functionals [47, 50, 55]. However, shape information was not used. Gharipour and Liew [52] employed level sets using local optimization. Individual cell nuclei were subsequently detected by determining concave points similarly to [32] and then delineated by a shortest-path method. Image intensities were not used for detection of individual cell nuclei.

To jointly exploit shape and intensity information, variational level sets were also used in conjunction with different kinds of shape regularization (e.g., [56, 57, 58, 59, 60, 61]). Ali and Madabhushi [56] employed elliptical shape priors based on signed distance functions for segmentation of cell nuclei in histological images. Lu et al. [61] proposed a shape prior based on the geometrical distance to the centroids of the cell nuclei, which were previously detected by connected component analysis. Kong et al. [57] and Zhang et al. [60] proposed similar approaches as [56] using more general shape priors, based on sparse linear combinations of priorly determined shape prototypes. Nosrati and Hamarneh [58] introduced a star-shape prior for the segmentation of the cytoplasm of cervical cells. Xing and Yang [59] exploited shape information to accelerate the evolution of similarly shaped level set functions, using a CNN for initialization (see Section 2.3.3). However, none of these methods yields globally optimal solutions and initialization is thus crucial.

Parametric shape models

The shape-constrained snakes [41, 42, 43, 44] described above can also be seen as a subset of the more general class of *explicitly parameterized shape models*. This class of models also includes non-snake based models (e.g., [62, 63, 64, 65, 66, 67, 68, 69]), with the used frameworks being the crucial difference.

In probabilistic frameworks, a *marked point process (MPP)* was used to represent an ensemble of multiple shape models (e.g., [62, 63, 64, 67]). An MPP is a random set of points (see, e.g., [70]), where each point is associated with the shape parameters and position of a shape model. The energy function is embedded into a probability density function and model fitting is performed by sampling from the density. Dong and Acton [62] employed a variant of Markov chain Monte Carlo sampling [71] based on elliptical models, and a pairwise interaction model to cope with cell clusters and partially overlapping objects. Soubies et al. [63] proposed a contrast-invariant energy function for elliptical models based on the image gradient, but did not exploit the image intensities. Sampling was performed using graph cuts [72, 73]. Descombes [64] embedded an MPP into a simulated annealing scheme. In principle, this scheme converges to a globally optimal solution, but it is computationally

intractable[1] in practice and requires careful calibration of the cooling parameters. Markowsky et al. [67] employed an MPP to determine a low-cardinality set of circles (formally corresponding to Definition 2.2) in order to split touching and overlapping cell nuclei in binarized images. Iteratively updated parameter distributions were used to guide the MPP towards a globally optimal solution, but the approximation ratio was not studied (cf. Section 2.2.6) and image intensities were not exploited.

Panagiotakis and Argyros [65] proposed a similar approach based on elliptical models and expectation maximization instead of MPP and sampling, which was later extended to better cope with partially overlapping objects [66]. However, only binarized images were used to separate individual objects. Ducroz et al. [68] and Eck et al. [69] used shape parameterizations based on spherical harmonics for analysis of cellular shapes [68] and segmentation of cellular structures [69], but globally optimal solutions were not obtained.

Compared to explicitly parameterized models, *implicit* parameterizations have analytic and algorithmic advantages (e.g., easier representation of closed curves, convenient representation by matrix-vector multiplication). In particular, implicitly defined shape models are computationally advantageous since energy minimization can often be performed by convex optimization. For example, Biesdorf et al. [75] proposed *implicitly* parameterized tubular shape models for vessel segmentation based on convex energy minimization. However, implicitly parameterized shape models were not used for cell nuclei segmentation.

Graph-based models

Discrete models are powerful because they permit determining solutions close to global optimality at low computational cost. In graph-based discrete models, vertices typically represent image points (e.g., [76, 77]) or objects (e.g., [78, 79, 80, 81]), and edges represent interactions. Segmentation can then be seen as vertex labeling, meaning that a discrete label is assigned to each vertex (e.g., an object label for image points or a boolean label indicating whether the object is included in a set). The Potts model [82] is a pairwise interaction model widely used in graph-based approaches,[2] which encourages vertex label consistency and penalizes label discontinuity. Using graph cuts [72] for energy minimization based on the Potts model yields an approximation of the globally optimal solution within an approximation guarantee (cf. Section 2.2.6) of factor 2 [84]. The globally optimal solution is exactly determined if only two labels are involved (e.g., image background and foreground).

[1] Descombes [64] reported "heavy computational time" but did not assess the computational cost quantitatively. Computationally favorable properties like convexity were not exploited (the energy function is non-convex). State-of-the-art simulated annealing methods like [74] demonstrate exponential run time. In view of Section 2.2.1, this suggests that the approach [64] is computationally intractable, which we also confirmed in our own experiments (Section 5.4.1).

[2] Graph-based models are not to be confused with *graphical models*, the latter being graph-based representations of joint probability distributions [22]. An example of a graphical model is obtained by embedding the Potts model into a Gibbs distribution [83].

Al-Kofahi et al. [76] used the Potts model and graph cuts for cell nuclei segmentation, also exploiting the four color theorem to reduce the number of labels and the computational cost. However, shape information is difficult to handle in a purely graph-based framework. For example, Lou et al. [77] employed blob detection to *a priori* detect individual cell nuclei. Then, graph cuts were used to solve a Potts model which favors cuts tangential to the previously detected blobs, thus implementing a blob-like shape prior. Poulain et al. [78] used a sequence of graph cuts to determine the best maximum subset of segmentation candidates formally corresponding to max-weight set-packing,[3] but properties of computational cost (cf. Section 2.2.1) and the approximation ratio were not studied (cf. Section 2.2.6).

For performing cell segmentation jointly with cell tracking using temporal image sequences (e.g., [79, 80, 81]), graphs were also used to represent an overcomplete set of trajectories (comprising correctly detected and spurious cell nuclei). To perform joint segmentation and tracking, the temporally most coherent subset of non-conflicting trajectories needs to be determined. This task is a variant of max-weight set-packing (Definition 2.3) and thus **NP**-hard. Akram et al. [79] proposed an approximation scheme based on sequential shortest path methods but did not investigate the approximation ratio. Akram et al. [80] and Türetken et al. [81] used integer linear programming to exactly determine the globally optimal solution at the price of high computational cost. Tracking-based methods are not applicable for the segmentation of individual images.

2.3.3 Learning-based approaches

Instead of using hand-tailored models to represent objects or images (Section 2.3.2), deep learning-based approaches are actively trained to *learn* representations [85]. Training, an immanent step of these approaches, is performed by minimization of a suitable loss function *prior* to application.

In recent years, *convolutional neural networks (CNN)* have become increasingly popular for cell segmentation in microscopy images (e.g., [86, 87, 88, 89, 90, 91, 92, 93, 94, 95, 96]). Ronneberger et al. [86] and He et al. [97] introduced the two predominant CNN architectures, U-Net and Mask R-CNN, respectively. Many extensions and variants were proposed to improve the segmentation performance. For example, Fan and Rittscher [89] and Vuola et al. [91] employed ensembles of multiple CNNs. To better cope with closely clustered or partially overlapping cell nuclei, Böhm et al. [88] and Payer et al. [92] used higher-dimensional embeddings, and Wollmann et al. [94] proposed a variant of the focal loss [98], which is sensitive to cell boundaries. Methods were also based on ensembles of CNNs and graphical models [87, 89, 93] or gradient flow fields [95, 99]. Fehri et al. [93] used Bayesian

[3] The objective of the optimization problem in [78] is to determine a minimum-weight subset of disjoint objects using non-positive weights. This is equivalent to determining a maximum-weight subset of disjoint objects using non-negative weights, which corresponds to Definition 2.3.

polytrees with learned deep features and globally optimal inference. However, training deep CNNs generally requires large amount of manually annotated data, and manual annotation of pixel-based segmentation masks is tedious. To cope with this, Xie et al. [90] used synthetic training data, and Ciga and Martel [96] used higher-level annotations for classification of image regions.

However, training deep neural networks is computationally expensive in general. In addition, it was shown that neural networks are prone to adversarial perturbations (small intensity fluctuations in the input images) [100], universal perturbations (e.g., random noise, geometric transformations) [101], and misannotated image data (label noise) when using deep high-capacity models [102]. Explainability, interpretability, and predictability of deep neural networks are open research topics.

Chapter 3

Datasets, baseline methods, and performance measures

3.1 Introduction

The proposed methods for cell segmentation (Chapters 4–6) were applied to fluorescence microscopy datasets of different cell types comprising various challenges, and we performed a quantitative comparison with previous methods. In this chapter, we describe the used datasets (Section 3.2), the previous methods used as baseline for the comparison (Section 3.3), and the performance measures used for quantification of the segmentation performance (Section 3.4).

3.2 Datasets

We applied our methods to seven fluorescence microscopy datasets in total, comprising different cell types, fluorophores, and challenges. The datasets are described below. An overview is given in Table 3.1.

			Number of	
Dataset (cell type)	Dye	Image resolution	Images	Cell nuclei
NIH3T3	Hoechst	1344×1024	49	2212
U2OS	Hoechst	1349×1024	48	1836
GOWT1 dataset 1	GFP	1024×1024	31	150
GOWT1 dataset 2	GFP	1024×1024	20	128
Fibroblast	DAPI	1024×1024	175	985
HeLa	DAPI	1200×1620	25	282
Macrophage	DAPI	1388×1040	20	700

Table 3.1. Overview of the used datasets. The image resolution is given in pixels. The number of cell nuclei corresponds to the number of annotated cell nuclei in the ground truth data.

- **NIH3T3 dataset [103].** This dataset consists of 49 images with a size of 1344 × 1024 pixels. The dataset includes 2212 cell nuclei in total, which were stained with Hoechst 33342, and is challenging because of strong autofluorescence artifacts, visible debris, significant intensity inhomogeneities, and many closely clustered or partially overlapping cell nuclei.

- **U2OS dataset [103].** The dataset consists of 48 images of U2OS cells that were stained with Hoechst 33342. The images have a size of 1349 × 1030 pixels and the dataset includes 1836 cell nuclei. The dataset is challenging due to frequent occurrence of strongly non-elliptical and closely clustered cell nuclei.

- **GOWT1 datasets [104].** We used two datasets of GFP-transfected mouse embryonic stem cells from the IEEE ISBI 2013 Cell Tracking Challenge [105] training data. The datasets are temporal image sequences, where each image has a size of 1024 × 1024 pixels. GOWT1 datasets 1 and 2 consist of 31 images (with 150 nuclei) and 20 images (128 nuclei), respectively. The two datasets are challenging due to low signal-to-noise ratio, low image contrast, and cell nucleoli (distinct dark regions within cell nuclei). The ground truth consists of eight fully annotated images and partial annotations for the other images.

- **Fibroblast dataset [106].** This dataset contains 175 3-D stacks of 35–50 DAPI-stained images of human fibroblast cells. Each image has a size of 1024 × 1024 pixels. The dataset includes 985 cell nuclei. For most of the 3-D stacks (112 out of 175), the cells were serum-starved and forced to arrest in the same cell cycle phase. From each of the 175 stacks, we used the image slice with the highest object density for the evaluation. The ground truth does not include objects at the image boundaries, causing wrong false-positive detections. To avoid misleading results, objects detected at image boundaries were ignored for Dice and Rand. The dataset is challenging due to partially strong intensity inhomogeneities of the cell nuclei.

- **HeLa dataset.** We also used 25 DAPI-stained HeLa images. The images contains 282 cell nuclei and each image is 1200 × 1620 pixels in size.

- **Macrophage dataset [107].** The dataset consists of 20 images of DAPI-stained murine bone-marrow derived macrophage cells. The images have a size of 1388 × 1040 pixels and show 30-50 cell nuclei per image. Half of the images are strongly out of focus, but blurring artifacts also exist in those images which are in focus.

3.3 Baseline methods

We used the following previous methods as baseline for our evaluation:

- **Global intensity thresholding (Otsu).** A global intensity threshold is computed from the image histogram [33]. Connected component analysis is employed to identify individual objects.
- **Blob detection-based level sets (Blob-LS).** The variational level set model of Chan and Vese [47] is used. For initialization, image blobs are detected by multi-scale Laplacian of Gaussian filtering [108]. Individual objects are determined using connected component analysis.
- **Blob detection-based random walker (Blob-RW).** First, a multi-scale Laplacian of Gaussian-based blob detector [108] is used. For each detected blob, a circular foreground marker is initialized (half the radius of the detected blob). The background marker is determined as the watershed of the negative image intensities between the foreground markers. Foreground and background markers are then expanded using the random walker algorithm [109].
- **Template matching (TM [110]).** First, supervised learning is performed to build a filter bank of templates, which is employed for the detection of cell nuclei. Non-rigid registration is then used for local alignment of the templates within each detected image region.
- **Graph cuts with blob-like shape prior (Blob-GC [77]).** First, the locations and sizes of cell nuclei are determined based on second-order image statistics. Segmentation is then performed by solving a Potts model, where cuts perpendicular to the object edges of the detected cell nuclei are penalized.
- **Two-stage thresholding (KTH [111]).** The two-stage thresholding method [29] is employed for segmentation. The connected components of the segmentation result are then used to perform cell tracking. The approach achieved the best result in the IEEE ISBI 2013 Cell Tracking Challenge [104].
- **Convex variational level sets (CVX-LS [51]).** The two-step approach exploits the convexity of level set functionals. First, an image is segmented and then semi-local refinement is performed. Shape information is not used.
- **Region-based progressive localization (RPL [112]).** Progressive contrast enhancement and pre-trained classifiers to detect salient image regions are used. Cell clusters are split using a binary classifier.
- **Blob detection-based approach (Blob-WS [79]).** Elliptical filter banks for nuclei detection and a watershed transform to obtain segmentation candidates are used. Temporal information from the image sequence is exploited to select or reject candidates.

- **Bayesian risk-based level sets (Bayes-LS [52]).** A level set functional based on the Bayesian classification risk is used, followed by morphological analysis to separate individual cell nuclei.
- **Cell Proposal Network (CPN [80]).** The method uses two convolutional neural networks (CNN). First, a CNN based on [113] is employed to determine candidates for cell nuclei bounding boxes. Then, another CNN based on the U-Net [86] is used for segmentation of each candidate. Finally, multiple hypothesis tracking is performed and the candidates corresponding to the most plausible trajectories are selected.
- **Ellipse-based shape decomposition (SEG-SELF [65], RFOVE [66]).** First, locally adaptive thresholding is used to binarize an image. Then, cell cluster splitting is performed by approximation of the binary image by a low-cardinality set of ellipses, either using hard (SEG-SELF) or soft (RFOVE) constraints for the overlap of the ellipses and the binary image.
- **Cellpose [95].** First, a modified U-Net [86] with residual blocks and global average pooling is used to predict a vector field. Then, individual objects are identified by grouping image points whose vectors point to the same location. The authors trained the model using a wide spectrum of different microscopy images including the NIH3T3 and U2OS datasets (Section 3.2).

3.4 Performance measures

The segmentation performance is assessed using measures that quantify the similarity between the segmentation results and manual ground truth annotations created by human experts. For our evaluation, we used region-based performance measures (Section 3.4.1), contour-based performance measures (Section 3.4.2), and detection-based performance measures (Section 3.4.3). These are detailed below.

3.4.1 Region-based performance measures

The used *region-based* performance measures are defined based on the set of all ground truth objects $\mathscr{R}_{\mathrm{gt}}$ within an image and the set of all segmented objects $\mathscr{R}_{\mathrm{seg}}$:

- **Dice similarity coefficient (Dice).** The Dice coefficient is defined as

$$\mathrm{Dice}\left(\mathscr{R}_{\mathrm{gt}}, \mathscr{R}_{\mathrm{seg}}\right) = 2 \cdot \frac{\#\left(\left(\bigcup \mathscr{R}_{\mathrm{gt}}\right) \cap \left(\bigcup \mathscr{R}_{\mathrm{seg}}\right)\right)}{\#\left(\bigcup \mathscr{R}_{\mathrm{gt}}\right) + \#\left(\bigcup \mathscr{R}_{\mathrm{seg}}\right)} \tag{3.1}$$

  and measures the overlap of the ground truth and the segmentation result, where 0 means no overlap and 1 means perfect agreement. Dice corresponds to the pixel-based F_1 score (harmonic mean of precision and recall).

- **Rand index (Rand).** The Rand index measures the similarity of the ground truth and the segmentation result [103]. Rand corresponds to the pixel-based accuracy score and, in contrast to Dice, is not biased towards positive or negative detections. A Rand value of 0 means no overlap, and a Rand value of 1 means perfect agreement.

- **Object-based Jaccard index (SEG).** In contrast to Dice and Rand, which only consider the union of all objects within an image, the SEG measure [104] takes into account the performance for individual objects. For each ground truth object $G \in \mathscr{R}_{\mathrm{gt}}$, the measure is defined as

$$\mathrm{SEG}\left(G, \mathscr{R}_{\mathrm{seg}}\right) = \begin{cases} \frac{\#(G \cap S)}{\#(G \cup S)} & \text{if } \exists S \in \mathscr{R}_{\mathrm{seg}} : \#(G \cap S) > 0.5 \cdot \#G, \\ 0 & \text{else} \end{cases}$$

  and attains values between 0 and 1. The SEG value is 0 if no segmented object overlaps the ground truth objects by at least 50 % (e.g., due to very inaccurate segmentations, falsely split/merged, or undetected objects). The SEG value is 1 if a ground truth object is perfectly segmented.

Notably, Dice and Rand are sensitive to false-positive detections, but invariant to falsely split/merged objects. On the other hand, SEG is sensitive to falsely split/merged objects and false-negative detections (but invariant to false-positive detections). Overall, SEG is the most comprehensive and best suited measure for region-based segmentation performance since it incorporates both detection and object-based segmentation performance.

3.4.2 Contour-based performance measures

We also used two *contour-based* performance measures. Both measures are based on the Euclidean distance $\mathrm{dist}_{\partial G}(x) = \min_{x' \in \partial G} \|x - x'\|$ of an image point x to the contour ∂G of the ground truth object $G \in \mathscr{R}_{\mathrm{gt}}$ and the corresponding segmented object $S \in \mathscr{R}_{\mathrm{seg}}$:

- **Object-based Hausdorff distance (HSD).** The Hausdorff distance

$$\mathrm{HSD}(G, S) = \max_{x \in \partial S} \mathrm{dist}_{\partial G}(x) \tag{3.2}$$

  is the maximum distance of the object contour ∂G to the contour ∂S of the segmented object [114]. The HSD is not upper-bounded and attains 0 if the two objects are identical.

- **Object-based normalized sum of distances (NSD).**
  The NSD measure is defined by

$$\mathrm{NSD}(G, S) = \sum_{x \in G \triangle S} \mathrm{dist}_{\partial G}(x) \,/\, \sum_{x \in G \cup S} \mathrm{dist}_{\partial G}(x), \tag{3.3}$$

  where $G \triangle S = (G \setminus S) \cup (S \setminus G)$ is the symmetric difference of G and S. The NSD is the ratio of the number of image points, which are either only in G or only in S, where each image point is weighted by its distance to the ground truth object contour [103]. The measure ranges from 0 (G and S are identical) to 1 (no overlap of G and S).

We computed HSD and NSD for all segmented objects of an image. In case of ambiguities, the correspondences between the ground truth objects $\mathscr{R}_{\mathrm{gt}}$ and the segmented objects $\mathscr{R}_{\mathrm{seg}}$ were established

1. either by choosing the segmented object $S \in \mathscr{R}_{\mathrm{gt}}$ which yields the largest overlap $\#(S \cap G)$ with the ground truth object $G \in \mathscr{R}_{\mathrm{gt}}$ (Chapter 5).

2. or by choosing the segmented object $S \in \mathscr{R}_{\mathrm{gt}}$ which is *closest* to the ground truth object $G \in \mathscr{R}_{\mathrm{gt}}$ using the respective distance function (Chapter 6).

3.4.3 Detection-based performance measures

In addition, we used two *detection-based* performance measures to better assess the segmentation performance of closely clustered or overlapping objects. We computed the average number of falsely merged (Merge) and split (Split) objects per image.

We used the method proposed in [103] to identify falsely merged and split objects. A bipartite graph was computed, comprising the segmented objects and the ground truth objects as the vertices of two independent sets (examples of such graphs are shown in Figure 3.1). A segmented object S and a ground truth object G are linked by an edge, if G is the ground truth object which the segmented object S has the largest intersection with (among all ground truth objects). Falsely merged and split objects are then easily identified by considering the *degree* of the vertices (i.e. the number of incident edges). A segmented object with degree two or larger corresponds to a *falsely merged* object (Figure 3.1a). Analogously, a ground truth object with degree two or larger corresponds to a *falsely split* object (Figure 3.1b). An object with degree one corresponds to a correctly merged and split object (Figure 3.1c).

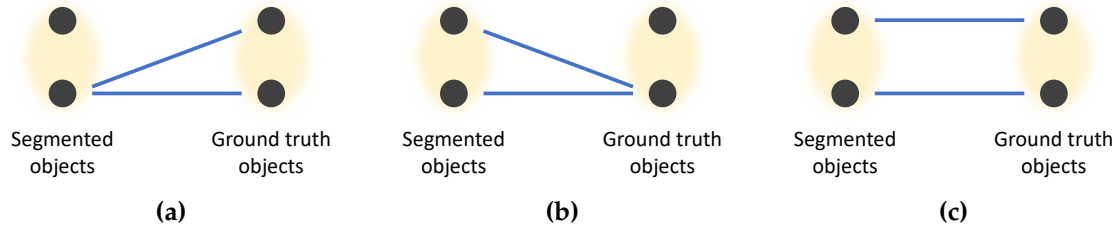


Figure 3.1. Bipartite graphs used for the detection-based measures. Vertices correspond to segmented (left) and ground truth objects (right). Edges indicate object pairs with maximum intersection. (a) One object is falsely merged. (b) One object is falsely split. (c) The two objects are correctly merged and split.

Chapter 4

Model fitting using elliptical models and sequential convex programming

4.1 Introduction

Key challenges of cell nuclei segmentation include strong image noise and the absence of distinct object boundaries, as depicted in Figure 4.1. In previous work, cell segmentation was often formulated a *model-based* energy minimization problem (cf. Section 2.3). Discrete models have the advantage that the solution can often be computed close to global optimality at low computational cost (e.g., [76, 77]). Such approaches are robust because they provably obtain one of the best admissible solutions. However, shape information is difficult to encode in discrete models.

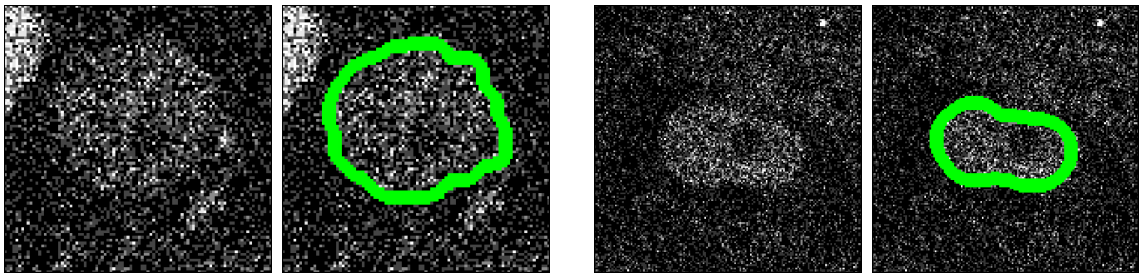


Figure 4.1. Two example images of GFP-transfected GOWT1 mouse embryonic stem cell nuclei (left and right) and corresponding ground truth segmentation.

Many cell segmentation methods are based on a *variational* framework (e.g., [48, 49, 50, 51, 52, 56, 57, 58, 59, 60]), where object contours are represented as level sets of functions. Formulating the evolution of such functions as a *convex* optimization problem (cf. Section 2.2.2) assures that a *globally optimal* solution is found reproducibly for any initialization (e.g., [51]). Exploiting *shape* information is advantageous to better cope with strong image noise and other distortions. Previously proposed shape-based approaches include shape-regularized variational level sets (e.g., [56, 57, 58, 59, 60]) or parametric shape models (e.g., [41, 42, 43, 44, 62, 63, 64, 65, 66, 67]). Model fitting was performed by probabilistic methods (e.g., [62, 63, 64, 67]), snake energy minimization (e.g., [41, 42, 43, 44]), or

expectation maximization (e.g., [65, 66]). None of these shape-based methods (except [64, 67]) yield *globally optimal* solutions. In [64], elliptical models are randomly sampled from uniform distributions, requiring a computationally intractable number of samples to obtain a globally optimal solution. The approach in [67] uses only circular models, requires a binarization of the image, and does not use the image intensities. None of the previous shape-based approaches for cell segmentation used convex optimization.

In this chapter, we introduce a new approach for cell nuclei segmentation, which is based on convex optimization and jointly exploits shape and intensity information. The approach uses implicitly parameterized elliptical shape models and we present a non-linear parameterization, which exploits the locations of priorly detected objects. The elliptical models are directly fitted to the image intensities, thus binarization of the image is not required. In our approach, the globally optimal minimizer of a suitable energy is estimated using a sequence of convex programs. A fast second-order method is employed to numerically solve each convex program. We also present a robust method for automatic selection of image regions, where model fitting is performed. Our approach is based on convexity and elliptical models and hence denoted CVXELL. We have evaluated CVXELL using fluorescence microscopy images of two different cell types and performed a quantitative comparison with previous methods.

The work has been published in Kostrykin et al. [11].

4.2 Approach

In this section, we describe our convex model-based approach for cell nuclei segmentation using elliptical shape models (CVXELL). Section 4.2.1 details the model and the optimization scheme based on sequential convex programming. Section 4.2.2 describes an approach for automatic selection of image regions for model fitting.

4.2.1 Model fitting by convex programming

Let Ω denote the set of all image points. We formulate the *shape model* as the zero-level set $C_s(\theta) = \left\{x \in \mathbb{R}^2 \middle| s(x;\theta) = 0\right\}$ of a θ-parametrized function $s : \Omega \to \mathbb{R}$, which maps an image point $x \in \Omega$ to a real value. With a symmetric matrix $A \in \mathbb{R}^{2\times 2}$, a vector $b \in \mathbb{R}^2$, and a scalar $c \in \mathbb{R}$, we choose the parameterization

$$s(x;\theta) = (x-b)^\top \cdot A \cdot (x-b) + c, \tag{4.1}$$

where the tuple $\theta = (A, b, c)$ is used for shorthand notation. Then, for a 2-D image, the zero-level set $C_s(\theta)$ is the whole image plane $\mathbb{R}^2$, the empty set, a single dot, or takes either an elliptic, hyperbolic, or linear shape.

The parameterization of the model function in Eq. (4.1) was previously used in [75] for vessel segmentation in 3-D CT data employing a first-order optimization scheme. In this work, we use the parameterization for segmentation of cell nuclei in fluorescence microscopy images. We also introduce different constraints and propose a much faster second-order optimization scheme (see below).

Energy formulation using a convex loss function

The model function in Eq. (4.1) induces the two disjoint image regions $\Omega_s^+(\theta) = \{x \in \Omega | s(x;\theta) > 0\}$ and $\Omega_s^-(\theta) = \{x \in \Omega | s(x;\theta) < 0\}$, which correspond to the zero-sublevel and zero-superlevel set of the function s, respectively, and identify the inside and the outside of the model if $C_s(\theta)$ is elliptical. Given an image $g\colon \Omega \to \mathbb{R}$, which depicts an object and its background in an image region $\omega \subseteq \Omega$, and an intensity offset τ, the *intensity model*

$$y(x) = g(x) - \tau, \tag{4.2}$$

induces the two regions $\Omega_y^+ = \{x \in \Omega | y(x) > 0\}$ and $\Omega_y^- = \{x \in \Omega | y(x) < 0\}$ analogously, corresponding to the imaged object and its background. To segment the object in the image region ω, we seek the model parameters θ so that $\omega \cap \Omega_s^+(\theta)$ matches $\omega \cap \Omega_y^+$ and $\omega \cap \Omega_{s_p}^-(\theta)$ matches $\omega \cap \Omega_y^-$. Formally, we minimize

$$\sum_{x \in \omega} L(y(x); s(x;\theta)), \qquad L(y;s) = \begin{cases} 1 & \text{if } y \cdot s < 0, \\ 0 & \text{else,} \end{cases} \tag{4.3}$$

where the $0/1$ loss function L penalizes each sample x with $\operatorname{sgn} y(x) \neq \operatorname{sgn} s(x;\theta)$. Since the loss function L in Eq. (4.3) is non-smooth, we instead use a surrogate loss function ϕ_γ (see, e.g., [23]) and consider

$$\sum_{x \in \omega} \phi_\gamma(y(x); s(x;\theta)), \qquad \phi_\gamma(y;s) = \ln(1 + \exp(-y \cdot s/\gamma)). \tag{4.4}$$

The function $\phi_\gamma / \ln 2$ is a *minimal* upper bound of L from Eq. (4.3), which also is smooth, and moreover *convex* in the model s (see Example 2.1, considering γ^{-1} as a factor of y). The factor $1/\ln 2$ is omitted in Eq. (4.4) because the minimizers of a function are invariant to positive constant factors. The value $\gamma > 0$ governs how strong samples x with $\operatorname{sgn} y(x) \neq \operatorname{sgn} s(x;\theta)$ are penalized. Since Eq. (4.4) is non-convex in the model parameters θ, we use an approximation scheme to estimate its global minimizer by solving a sequence of convex programs, as detailed below.

Sequential convex programming

Using $\theta = \theta_0 + \theta_\delta$, where θ_0 is a current estimate of the minimizer and θ_δ is an increment, leads to the decomposition $s(x;\theta) = s(x;\theta_0) + d(x;\theta_\delta) + d'(x;\theta_\delta)$ of

Eq. (4.1) with

$$d\left(x;\theta_\delta\right) = \left\langle A_\delta,\, x\cdot x^\top - 2x\cdot b_0^\top + b_0\cdot b_0^\top \right\rangle + \left\langle 2b_\delta,\, A_0\cdot(b_0 - x)\right\rangle + c_\delta \tag{4.5a}$$

being linear in θ_δ, and a higher-order term

$$d'\left(x;\theta_\delta\right) = \left\langle b_\delta,\, (A_0 + A_\delta)\cdot b_\delta\right\rangle + \left\langle 2b_\delta,\, A_\delta\cdot(b_0 - x)\right\rangle. \tag{4.5b}$$

Then, within the trust region $\|A_\delta\| \leq \epsilon_1$ and $\|b_\delta\| \leq \epsilon_2$ with small $\epsilon_1, \epsilon_2 > 0$ we have $d'\left(x;\theta_\delta\right) \approx 0$ and hence $s\left(x;\theta\right) \approx s\left(x;\theta_0\right) + d\left(x;\theta_\delta\right)$. This enables us to estimate the global minimizer of Eq. (4.4) by a sequential scheme: Keeping θ_0 fixed, we compute the increment θ_δ which globally minimizes the convex energy

$$\sum_{x\in\omega} f_x\left(\theta\right), \qquad f_x\left(\theta\right) = \phi_\gamma\big(y\left(x\right); s\left(x;\theta_0\right) + d\left(x;\theta_\delta\right)\big) \tag{4.6}$$

subject to the two *trust-region* constraints $\|A_\delta\| \leq \epsilon_1$ and $\|b_\delta\| \leq \epsilon_2$ (cf. Section 6.1.3 in [15]), which leads to $\theta_0 \leftarrow \theta_0 + \theta_\delta$ for the next iteration.

The energy in Eq. (4.6) is convex, since $\phi_\gamma\left(y;s\right)$ is convex in s, and $s\left(x;\theta_0\right) + d\left(x;\theta_\delta\right)$ is linear in p_δ (see the rule for convex-affine composition in Section 2.2.2). In our experiments we found that constraining $C_s\left(\theta\right)$ to be located within a predefined image region increases the robustness. With b as the center of an elliptic shape model $C_s\left(\theta\right)$, we formulate a convex program that is solved in each iteration and constrains b to stay within a maximum distance $r \in \mathbb{R}_+$ of a preset point $\mu \in \mathbb{R}^2$:

$$\begin{aligned} \text{minimize} \quad & \sum_{x\in\omega} f_x\left(\theta\right) \\ \text{subject to} \quad & \|A_\delta\| \leq \epsilon_1, \\ & \|b_\delta\| \leq \epsilon_2, \\ & \|b_\delta + b_0 - \mu\| \leq r, \end{aligned} \tag{4.7}$$

The feasible set in Eq. (4.7) is convex because the constraints are determined by convex functions (norms) of affine mappings. These constraints are different to [75], where 3-D tubular shapes were *enforced* for vessel segmentation in 3-D CT data and location constraints were not used. In contrast, we *avoid* fitting an elliptical model when no such structure is present.

The global solution of the constrained convex optimization problem in Eq. (4.7) is found by primal-dual interior-point methods (see Section 2.2.5), for which we used the implementation [115]. This second-order optimization scheme is significantly faster than using a first-order method to solve Eq. (4.7) directly. The initialization $\theta_\delta = 0$ is always feasible if $\|b_0 - \mu\| \leq r$ is assured. We incrementally refine the estimate θ_0 by solving the optimization problem in Eq. (4.7), until the increment $\|\theta_\delta\|$ becomes smaller than $\epsilon_{\min}$ or $n_{\max}$ iterations are reached. This is outlined in Algorithm 4.1, which determines the solution $\hat{\theta} = (\hat{A}, \hat{b}, \hat{c})$.

Algorithm 4.1: Intensity-based model fitting using sequential convex programming and location constraints.

input: Ω, g, τ, γ, μ, r, $n_{\max}$, $\epsilon_{\min}$

Initialize $A_0 \leftarrow \left[\begin{smallmatrix} -1/r^2 & 0 \\ 0 & -1/r^2 \end{smallmatrix}\right]$; $b_0 \leftarrow \mu$; $c_0 \leftarrow 1$;

for $i \leftarrow 1$ **to** $n_{\max}$ **do**

  Set $\theta_\delta \leftarrow$ solution of Eq. (4.7) using the initialization $\theta_\delta = 0$;

  if $\|\theta_\delta\| < \epsilon_{\min}$ **then break**;

  Update $\theta_0 \leftarrow \theta_0 + \theta_\delta$;

return θ_0;

4.2.2 Selection of image regions

Our approach presented above determines the shape of a cell nucleus within an image region ω. Since the shape of $C_s(\hat{\theta})$ is not restricted to ellipses by the constraints of Eq. (4.7), the result of Algorithm 4.1 satisfies

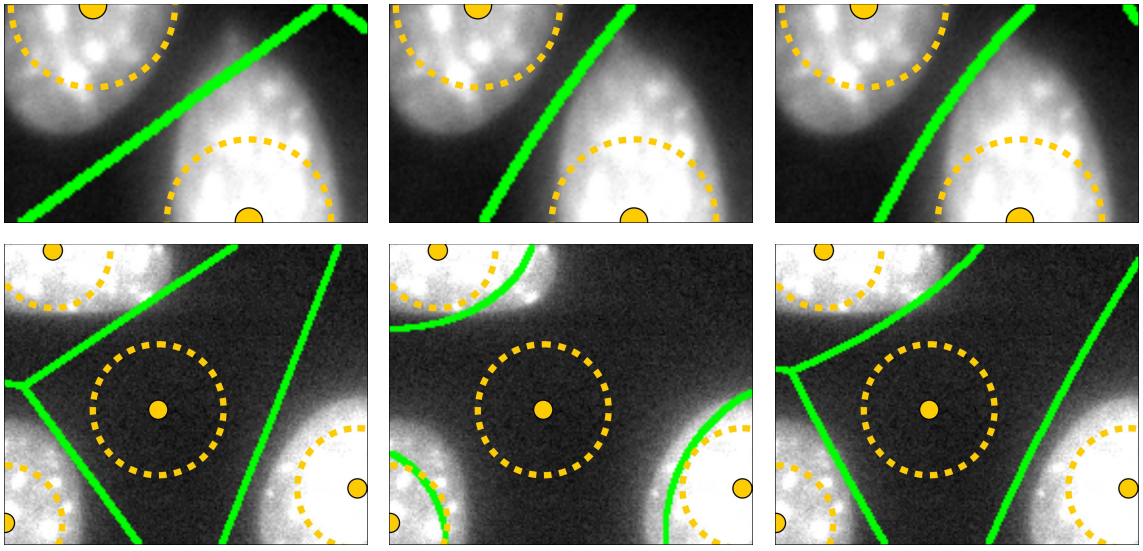
$$\hat{A} \prec 0 \quad \wedge \quad \hat{c} > 0 \tag{4.8}$$

if and only if an elliptical structure is present in ω, where "$\prec$" denotes the partial order with respect to the semidefinite cone. Thus, our aim is to determine *one* region ω for each cell nucleus in the image, where falsely detected regions without cell nuclei are tolerable, because using Eq. (4.8) for testing allows identifying and discarding empty regions in a simple but reliable manner.

To determine suitable image regions, we employed a multiscale blob detector [108]. We used a very conservative threshold for the detection, since falsely detected blobs (and the corresponding fitted shapes) can be reliably discarded by Eq. (4.8). Densely located cell nuclei were handled by applying the detector not directly to the image intensities g, but to the image $g - \|\nabla g\| / \max_x \|\nabla g\,(x)\|$, where $\|\nabla g\|$ is the gradient magnitude of g. The image is then partitioned into distinct regions by assigning each pixel x to the i-th blob, for which the squared Mahalanobis distance $(x - \mu_i)^\top \cdot D_i \cdot (x - \mu_i)$ to the center μ_i of the i-th blob is minimal. We used the matrix

$$D_i = -H_i \,/\left(\sigma_i^2 \cdot \lambda_{\min}\left(-H_i\right)\right), \tag{4.9}$$

where H_i is the Hessian matrix of the image intensities g at μ_i and $\lambda_{\min}\left(-H_i\right)$ is the lowest eigenvalue of $-H_i$. Blobs with $\lambda_{\min}\left(-H_i\right) = 0$ were not considered. The denominator in Eq. (4.9) normalizes the scale information in H_i with respect to the blob scale σ_i. Using the Hessian matrix of an image for analyzing local structure is not new (e.g., [116]). However, in our approach we use the Mahalanobis distance and the normalization in Eq. (4.9). Figure 4.2 shows that using the Euclidean distance tends to split cell nuclei into multiple regions (first row), the Mahalanobis



(d) (e) (f)

Figure 4.2. Voronoi diagrams (green) of detected blobs (yellow dot and circle) using (a) Euclidean distance ($D_i = I$), (b) Mahalanobis distance ($D_i = -H_i$), (c) Mahalanobis distance using the normalization in Eq. (4.9) for two example images (top, bottom) from the NIH3T3 dataset.

distance without normalization is prone to falsely-detected blobs (second row), and our approach performs best.

For each determined region ω, we estimated the globally optimal fit of our shape model using Algorithm 4.1. The parameters μ and r were chosen as the center and radius $\sqrt{2} \cdot \sigma_i$ of the blob, the threshold τ was determined by Otsu thresholding [33] of the region ω. We used $\gamma = 0.025$, $n_{\max} = 50$, $\epsilon_{\min} = 0.1$, $\epsilon_1 = 20$, and $\epsilon_2 = 1$ in all our experiments. For every fitted model passing the condition in Eq. (4.8), we further examined the eigenvalue decomposition $\hat{A} = V \cdot \Lambda \cdot V^\top$ and computed the ellipse half-axes $[l_1, l_2] = -\sqrt{\hat{c}} \cdot \Lambda^{-1/2} \cdot V$. We discarded those results additionally, for which the ellipse area $\pi \cdot \|l_1\| \cdot \|l_2\|$ or circularity $\|l_1\| / \|l_2\|$ significantly differed from the means of the same image, and merged ellipses with sufficient overlap into single objects.

4.3 Evaluation

First, we studied the convergence properties of our CVXELL approach, which leverages a second-order optimization scheme. Using the image shown in Figure 4.3a, we compared our method with the first-order scheme in [75], which was previously used for vessel segmentation in 3-D CT data, but omitting the tubular shape constraints. Our approach converged in 0.6 seconds (after five iterations there were hardly any changes), achieving a Dice coefficient of 84% (see Figure 4.3d). In comparison, the scheme in [75] required already 3.3 seconds for one iteration

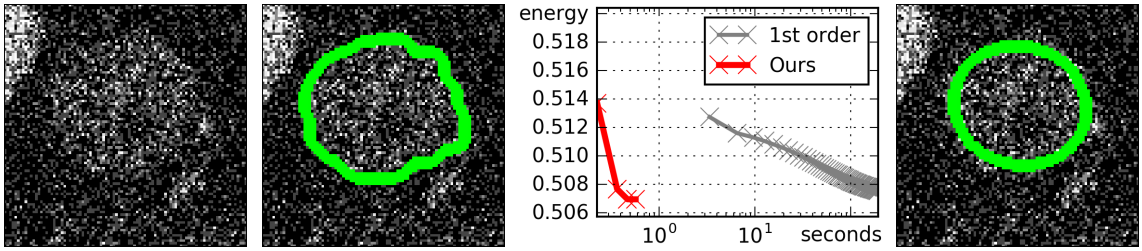


Figure 4.3. Example of the performance of our CVXELL approach. (a) Section of an example image from GOWT1 dataset 1. (b) Ground truth. (c) Convergence of our second-order approach (red) compared to first-order optimization (gray). (d) Our segmentation result.

(Dice coefficient: 80%) and did not terminate within 100 seconds (using the same parameter settings as for our approach).

Second, we applied CVXELL to images from three datasets. The first dataset (from [103]) consists of 49 images of Hoechst-stained NIH3T3 cells. Visible artifacts and non-elliptic nuclei shapes hamper the analysis of these images. The other two datasets are training datasets from the ISBI 2013 Cell Tracking Challenge [104], consisting of 51 images of GFP-transfected GOWT1 cells. Both datasets are difficult due to strong image noise and low contrast (e.g., see Figure 4.3a). A detailed description of the datasets is given in Section 3.2.

We used the Dice coefficient and the SEG measure from [104] to evaluate our results. SEG was computed for all images. It is defined as the mean Jaccard similarity index $J(R) = |R \cap S(R)| / |R \cup S(R)|$ of a ground truth cell nucleus R and its corresponding segmented object $S(R)$. If no segmented object corresponds to R, $J(R) = 0$ is set. For the Dice coefficient, we used all images, which a fully-labeled ground truth was available for (all 49 images for the NIH3T3 dataset and 4 images for each GOWT1 dataset). See Section 3.4 for details.

For the evaluation with the Dice coefficient, we used the NIH3T3 dataset and both GOWT1 datasets. For NIH3T3, results were previously reported for CVX-LS [51], which is a variational level sets approach based on convex energies but shape information was not used (see Section 3.3). We performed a comparison with this method and Otsu thresholding, and studied the effectiveness of the location constraint of our approach in Eq. (4.7). The results in Table 4.1 show that the location constraint improves the accuracy significantly. In addition, our CVXELL approach outperforms CVX-LS [51] and Otsu thresholding.

We also assessed the performance of our method on the two GOWT1 datasets using the SEG measure and compared it with three other methods, for which [80] provided results. The first method is KTH [111], which performed overall best for segmentation in the ISBI 2013 Cell Tracking Challenge [104]. The other two methods are a blob detection approach (Blob-WS) [79] and a deep learning method

Dataset	Otsu	CVX-LS	CVXELL w/o LC	CVXELL
NIH3T3	40.5 %	85 %	86.4 %	**87.4 %**
GOWT1 dataset 1	59.2 %	—	57.4 %	**63.7 %**
GOWT1 dataset 2	60.4 %	—	86.7 %	**89.4 %**

Table 4.1. Average Dice coefficients of our CVXELL approach with and without location constraints (LC) and other approaches. The best-performing method of each dataset is highlighted.

(CPN) [80]. For a description of the methods see Section 3.3. We also used Otsu thresholding for our comparison. The results in Table 4.2 show that our method performs best on the second GOWT1 dataset. For the first GOWT1 dataset, our method is second best (somewhat worse than CPN), but significantly more accurate (13.6 %P[4] better) than KTH, which achieved the best overall result for segmentation in the challenge [104]. Our method is also 7.9 %P better than Blob-WS. The low values for Otsu thresholding indicate the difficulties of the datasets, where cell nuclei are easily missed due to low contrast and strong noise. Considering this, and also that BLOB and particularly CPN exploit temporal information by performing joint segmentation and tracking, which our method does not, the results of our method are very competitive.

Dataset	Otsu	CPN	Blob-WS	KTH	CVXELL
GOWT1 dataset 1	21.7 %	**85.1 %**	74.2 %	68.5 %	82.1 %
GOWT1 dataset 2	42.5 %	87.3 %	90.5 %	89.4 %	**91.3 %**

Table 4.2. SEG performance values of Otsu thresholding, CPN, Blob-WS, KTH, and our CVXELL approach. The best-performing method of each dataset is highlighted.

4.4 Discussion

In this chapter, we have presented a new model-based approach for robust segmentation of cell nuclei in microscopy images. An elliptical shape model is directly fitted to the image intensities by solving a sequence of convex programs, thus, shape and intensity information are jointly exploited. A fast second-order optimization scheme determines the global solution of each convex program. By the choice of the constraints in the convex programs, our approach is intrinsically tolerant to falsely selected image regions. A quantitative comparison with previous methods showed that our approach yields competitive results or outperforms previous methods.

[4] %P denotes percentage points.

Chapter 5

Joint segmentation and cluster splitting using elliptical models

5.1 Introduction

A central problem in cell segmentation is coping with closely clustered or partially overlapping objects. To address this, approaches that exploit shape information have been introduced (for a review see Section 2.3). These shape-based approaches often perform segmentation and separation of individual objects (cell cluster splitting) *consecutively*. There are two main schemes:

1. **Binarization-based.** In this scheme, *image binarization* is performed first and individual objects are delineated subsequently by analysis of the binary image (e.g., [32, 52, 65, 66, 67]). Methods for binary image analysis include morphological analysis [32, 52], probabilistic methods [67], and expectation maximization [65, 66]. A disadvantage of these approaches is that shape and intensity information are only used in *consecutive* steps, but not *jointly*. Individual objects are identified using only the initial binarized image.

2. **Detection-based.** In this scheme, *prior object detection* is performed to identify individual cells before segmentation (e.g., [58, 61, 77]). Object detection was performed either by analysis of local second-order image statistics (e.g., [77] and Chapter 4), using random decision forests [58], and connected component analysis [61]. These approaches *jointly* exploit shape and intensity information, but they heavily depend on the result of the initial object detection, which determines how cell clusters are eventually split.

Active contours based on *snakes* (e.g., [41, 42, 43, 44]) or variational level sets with *shape priors* (e.g., [56, 57, 59, 117]) can be seen as a sub-group of the second scheme (detection-based). The reason is that accurate initialization is crucial for these

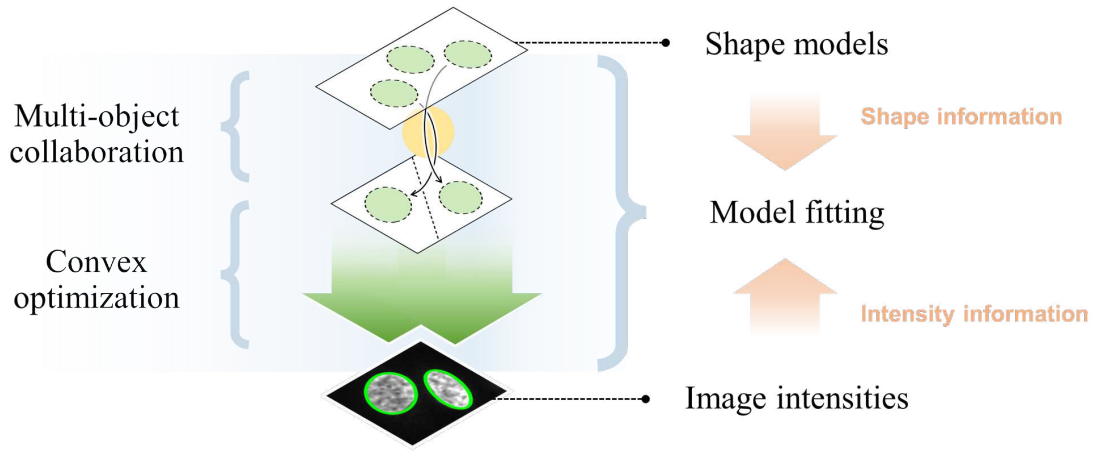


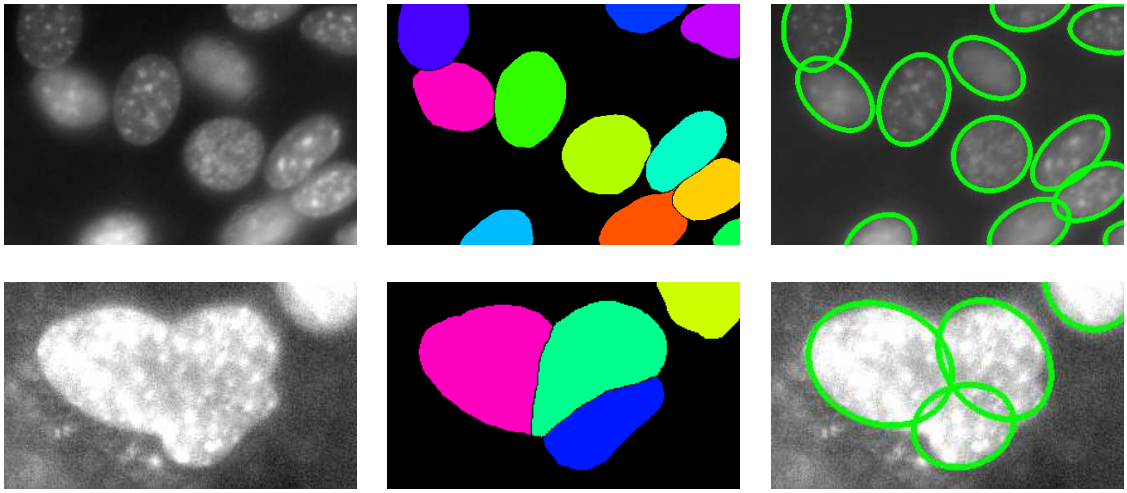
Figure 5.1. Overview of our GOCELL approach for cell nuclei segmentation.

approaches, since they do not yield a globally optimal solution, and initialization requires detection of the individual cell nuclei.

Shape-based approaches without requiring prior image binarization or object detection have also been proposed (e.g., [62, 63, 64]) which were based on probabilistic methods for model fitting. However, the methods of Dong and Acton [62] and Soubies et al. [63] do not obtain globally optimal solutions. Descombes [64] described an approach based on simulated annealing, which is computationally intractable in practice and requires careful calibration of the cooling parameters.

In this chapter, we introduce a new *globally optimal* approach for cell nuclei segmentation, which *jointly* exploits shape and intensity information. The approach is based on implicitly parameterized elliptical models and global energy minimization. Our proposed shape parameterization is linear and leads to a convex energy for single objects, that is optimally minimized using robust numerical methods. The optimization does not depend on the initialization and does not suffer from local minima. To avoid prior detection of image regions corresponding to the individual cell nuclei, we generalize the single-object model to the multi-object case. Our multi-object model consists of multiple collaborating ellipses, which represent a whole image. This leads to a non-convex energy, yet we have found that model fitting using the multi-object model corresponds to the *min-weight set-cover* problem (cf. Definition 2.2 in Section 2.2.6). Assuming that individual objects are roughly elliptical, simply connected, and correspond to one or more local intensity peaks, the result is determined close to the *global solution* using an efficient combination of combinatorial and second-order convex optimization schemes. Figure 5.1 shows an overview of the proposed GOCELL (globally optimal collaborating ellipses) approach.

The proposed energies are contrast-invariant and thus our approach is robust to *inter-object* intensity inhomogeneities. The joint exploitation of shape and intensity information enables our approach to intrinsically cope with *intra-object* intensity inhomogeneities and partial object overlap (see Figure 5.2). Thus, for



(a) (b) (c)

Figure 5.2. Separation of touching (first row) and overlapping (second row) cell nuclei. (a) Original image section (NIH3T3 cells). (b) Corresponding ground truth data. (c) Segmentation result of the proposed approach (green contour).

splitting of clustered cell nuclei, our approach neither requires an object interaction model (e.g., [40, 49, 62, 63, 64, 67, 77, 78]) nor prior image binarization (e.g., [32, 52, 65, 66, 67]). In contrast to our previous single-object approach CVXELL (Chapter 4), the proposed single-object scheme leads to a convex energy, which can be directly (exactly) minimized and does not require an approximation. In addition, we propose a multi-object scheme. This scheme *jointly* performs cell segmentation and cluster splitting, avoiding the necessity for prior object detection (e.g., [41, 42, 43, 44, 56, 57, 58, 59, 61, 77, 117]). The structure of our approach inherently permits effective parallelization. In contrast to learning-based approaches (Section 2.3.3), our approach does not require data-driven training nor annotated data. A main advantage of our model-based approach is that the explicit model assumptions allow designing well-defined algorithms that facilitate both reproducibility and predictability. To the best of our knowledge, we propose the first globally optimal model-based approach which jointly exploits shape and intensity information and is computationally tractable in practical applications.

We have evaluated our approach using fluorescence microscopy datasets of five different cell types, including publicly available benchmark datasets, and performed a quantitative comparison with previous methods. It turned out that the proposed approach generally improves the performance.

This chapter is organized as follows. Section 5.2 presents our intensity-based segmentation approach using convex energies and an implicitly parameterized single-object model. Section 5.3 describes the generalized multi-object model and the corresponding scheme for global energy minimization. Experimental results

and a comparison with previous methods are provided in Section 5.4. We discuss the results of our work in Section 5.5.

The work has been published in Kostrykin et al. [12].

5.2 Single-object model and convex energy formulation

In this section, we describe our globally optimal approach for cell nuclei segmentation, which jointly uses shape and intensity information based on a single-object model. We define a *shape model* as the zero-level set $C_s(\theta) = \left\{x \in \mathbb{R}^2 \middle| s(x;\theta) = 0\right\}$ of a model function s, which maps an image point to a real value. More specifically, we parameterize s as a second-order polynomial,

$$s(x;\theta) = x^\top \cdot A \cdot x + b^\top \cdot x + c, \tag{5.1}$$

where the symmetric 2×2 matrix A, the vector $b \in \mathbb{R}^2$, and $c \in \mathbb{R}$ are represented by the shape parameter vector θ. The shape of the zero-level set $C_s(\theta)$ is then confined to an ellipse, a parabola, hyperbola, line, or a stripe, unless $C_s(\theta)$ corresponds to the whole image plane, the empty set, or a single dot. Given the set of all image points Ω, the model function s induces two disjoint image regions, which are the zero-sublevel set $\Omega_s^-(\theta) = \{x \in \Omega | s(x;\theta) < 0\}$ of s and its zero-superlevel set $\Omega_s^+(\theta) = \{x \in \Omega | s(x;\theta) > 0\}$. If the parameters θ are chosen so that the shape model $C_s(\theta)$ is elliptical, then the regions $\Omega_s^+(\theta)$ and $\Omega_s^-(\theta)$ correspond to the interior and exterior of the ellipse, respectively.

Segmentation using the parameterization in Eq. (5.1) is analogous to using the parameterization in Eq. (4.1) for CVXELL in Section 4.2.1, but there are important differences since Eq. (5.1) is *linear* in the shape parameters θ in contrast to Eq. (4.1) which is non-linear. A main advantage of using a linear parameterization is that it directly leads to a convex energy (see below).

Given an image $g\colon \Omega \to \mathbb{R}$ of an object and its background in an image region $\omega \subseteq \Omega$, which are *roughly* separable using an intensity offset τ, the τ-superlevel set $\Omega_{g-\tau}^+ = \{x \in \Omega | g(x) - \tau > 0\}$ indicates the imaged object, whereas the τ-sublevel set $\Omega_{g-\tau}^- = \{x \in \Omega | g(x) - \tau < 0\}$ corresponds to the image background. To segment the image region ω, we consider the *intensity* model $y(x) = g(x) - \tau$ in Eq. (4.2) and seek those shape parameters $\hat{\theta}$, for which $\omega \cap \Omega_s^+(\hat{\theta})$ covers $\omega \cap \Omega_y^+$ while $\omega \cap \Omega_s^-(\hat{\theta})$ covers $\omega \cap \Omega_y^-$. More formally, we minimize

$$\psi_{\omega,L}(\theta) = \sum_{x\in\omega} L(y(x); s(x;\theta)), \qquad L(y;s) = \begin{cases} 1 & \text{if } y \cdot s < 0 \\ 0 & \text{else,} \end{cases} \tag{5.2}$$

which penalizes each image point x with $\operatorname{sgn} y(x) \neq \operatorname{sgn} s(x;\theta)$. Since the energy $\psi_{\omega,L}$ is non-smooth, we instead determine the optimal shape parameters $\hat{\theta}$ as the

global minimizer of the energy function

$$\psi_\omega(\theta) = \psi_{\omega,\phi}(\theta) = \sum_{x\in\omega} \phi_\gamma\left(y(x); s(x;\theta)\right), \qquad \phi_\gamma(y;s) = \ln\left(1+\exp\left(-y\cdot s/\gamma\right)\right), \tag{5.3}$$

where the surrogate loss function ϕ_γ from Eq. (4.4) is smooth and convex in s and $\gamma > 0$ is a fixed factor. In addition, ϕ_γ is a minimal convex upper bound of the 0/1-loss L in Eq. (5.2), if L is weighted by the constant factor $\phi_\gamma(y;0) = \ln 2$. Thus, the minimization of the energy ψ_ω also minimizes the energy $\psi_{\omega,L}$, since the minimizers of a function are invariant to positive constant factors ($\ln 2$). Our approach exploits both shape and image intensity information and an extension to other image features (e.g., texture) is possible by including additional terms in y.

Since the proposed parameterization in Eq. (5.1) is *linear* in the shape parameters θ, the energy function ψ_ω in Eq. (5.3) is a sum of *convex* functions (cf. convex-affine composition in Section 2.2.2) and thus also convex (cf. convex weighted sum in Section 2.2.2). The energy formulation in Eq. (5.3) is analogous to logistic regression using polynomial basis function expansion (cf. Example 2.3 using $\Phi(x) = \left(x_1^2, x_2^2, 2x_1x_2, x_1, x_2, 1\right)$), but in contrast to logistic regression, the codomain of y in our approach is not limited to binary values.

In our previous CVXELL approach (Chapter 4), a different parameterization of the model function was used, which is not linear in the shape parameters θ (cf. Eq. (4.1)), and thus the corresponding energy is *non-convex*. There, we hence used a sequential approximation scheme for global energy minimization. In contrast, for the new parameterization in Eq. (5.1), the energy ψ_ω is *convex* and can be directly globally minimized without requiring an approximation. In our proposed approach, we determine the globally optimal parameters $\hat{\theta}$ by robust numerical methods and an arbitrary initialization, as detailed in Section 5.3.3.

Another advantage of the new parameterization in Eq. (5.1) is that it is homogeneous. Consequently, the optimal zero-level set $C_s(\hat{\theta})$ is invariant to the factor γ, since the feasible set of the parameters θ is unbounded, and thus closed under scalar multiplication. Using the parameterization in Eq. (5.1) we can assume $\gamma = 1$ without loss of generality. In our notation, we thus skip the dependence on γ below and define $\phi(y;s) = \phi_1(y;s)$ to improve readability. The factor $\gamma > 0$ in Eq. (5.3) linearly scales the intensity model y and thus governs the contrast between the imaged object and the background. Model fitting using the energy ψ_ω and the new parameterization in Eq. (5.1) is hence invariant to the image contrast.

5.3 Multi-object model and globally optimal energy minimization

The single-object model described above represents a single elliptical object. Hence, the model can only be fitted to an image region, which contains at maximum one single object. However, since cell microscopy images generally contain multiple objects, we generalize the single-object model in Eq. (5.1) to the multi-object case. Below, we describe the *multi-object* model, the method for global energy minimization, and implementation details.

5.3.1 Multi-object model

Recall that the single-object model function s, described in the Section 5.2 above, induces two disjoint image regions, which are defined by the zero-superlevel set $\Omega_s^+(\theta)$ and the zero-sublevel set $\Omega_s^-(\theta)$. If the zero-level set $C_s(\theta)$ has an elliptic shape, then the two regions correspond to the interior and exterior of the ellipse. For the multi-object case, we extend the model function s so that it represents *multiple* elliptical objects.

In the multi-object case, we seek to cover the image points $x \in \omega$ of the image foreground ($y(x) > 0$) by the *union* of the foreground of multiple models $s\left(\theta^{(1)}\right), \ldots, s\left(\theta^{(m)}\right)$ of the form in Eq. (5.1). At the same time, the image background (image points x with $y(x) < 0$) is covered by the *intersection* of the background of these models. The union of the foreground of the models $s\left(\theta^{(1)}\right), \ldots, s\left(\theta^{(m)}\right)$ can be expressed as $\bigcup_{k\in[m]} \Omega_s^+(\theta_k) = \Omega_{\tilde{s}}^+(\theta)$, which is the zero-superlevel set of

$$\tilde{s}(x;\theta) = \max_{k\in[m]} s\left(x;\theta^{(k)}\right) \qquad \text{where } \theta = \left(\theta^{(1)}, \ldots, \theta^{(m)}\right), \tag{5.4}$$

since, for fixed x and θ, $\tilde{s}(x;\theta) > 0$ occurs if and only if there is a $k \in [m]$ with $s\left(x;\theta^{(k)}\right) > 0$. The intersection of the background of the models is given by $\bigcap_{k\in[m]} \Omega_s^-\left(\theta^{(k)}\right) = \Omega_{\tilde{s}}^-(\theta)$, since $s(x;\theta) < 0$ occurs if and only if $s\left(x;\theta^{(k)}\right) < 0$ for all $k \in [m]$. Using the formulation in Eq. (5.4), the models $s\left(\theta^{(1)}\right), \ldots, s\left(\theta^{(m)}\right)$ thus *collaboratively* represent the image foreground and background. Below, to improve the readability, we define $s_k(x;\theta) = s\left(x;\theta^{(k)}\right)$ and skip the explicit dependence of $\tilde{s}$ and s_k on θ.

Model activity regions and region fragments

An example illustrating the multi-object model is provided in Figure 5.3. Naturally, at any given image point x, the pointwise maximum in Eq. (5.4) does not depend on models s_k with $s_k(x) < \tilde{s}(x)$. Hence, models s_k with $s_k(x) = \tilde{s}(x)$ are of major interest. In our approach, such a model is denoted to be *active* at x. The set of all image points, where this model is active, forms an *activity region*. Closer characterization

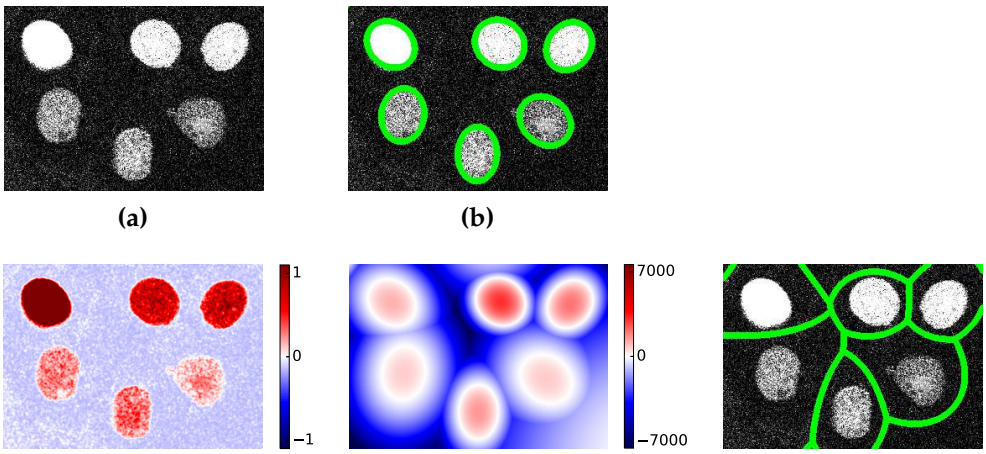


Figure 5.3. Example illustrating the multi-object model with optimally chosen model parameters $\hat{\theta}$. (a) Original image section (GOWT1 cells). (b) Zero-level set $C_s(\hat{\theta})$ (green) of the multi-object model s. (c) Intensity model $y\,(x)$ as a function of x. (d) Multi-object model $s(x;\hat{\theta})$ as a function of x. (e) Corresponding model activity regions (green contour).

of these regions proves to be advantageous, as detailed in Section 5.3.2 below. Figure 5.3e shows the activity regions, which correspond to the multi-object model depicted in Figure 5.3d.

Multi-object energy formulation

The multi-object model in Eq. (5.4) is homogeneous in the parameters θ, since the pointwise maximum and s_k are homogeneous functions. Thus, the multi-object model preserves the contrast invariance property of the single-object model and we may hence assume $\gamma = 1$ for the energy function of the multi-object model in Eq. (5.4) without loss of generality. Analogously to the segmentation of an image region ω (Section 5.2), segmentation of an entire image Ω using the multi-object model in Eq. (5.4) is performed by determining those shape parameters $\hat{\theta}$, for which $\Omega_s^+(\hat{\theta})$ covers Ω_y^+ while $\Omega_s^-(\hat{\theta})$ covers Ω_y^-. This corresponds to minimization of the energy function

$$\psi\,(\theta) = \sum_{x\in\Omega} \phi\Big(y\,(x)\,; \max_{k\in[m]} s_k\,(x)\Big), \tag{5.5}$$

where the image intensities are incorporated via the intensity model y in Eq. (4.2) using an intensity offset τ. The offset is computed by analyzing the local image intensities, as detailed in Section 5.3.3 below. Image binarization is not required.

Minimization of the energy function in Eq. (5.5) determines the parameters θ of the multi-object model in Eq. (5.4) which define the *globally optimal collaborating*

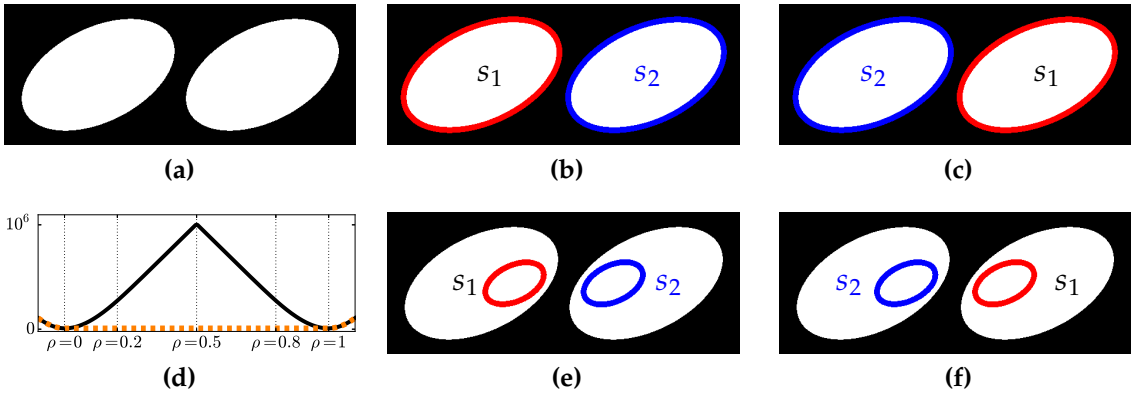


Figure 5.4. Example illustrating the non-convexity of the energy in Eq. (5.5) using $m = 2$ object models. (a) Synthetic image. (b) Parameters θ_0 corresponding to the zero-level sets of the two models (red for s_1 and blue for s_2). (c) Parameters θ_1, where $\theta_1^{(1)} = \theta_0^{(2)}$ and $\theta_1^{(2)} = \theta_0^{(1)}$. (d) Energy function from Eq. (5.5) along the ρ-parameterized line $\theta_\rho = \theta_0 \cdot (1 - \rho) + \theta_1 \cdot \rho$ between θ_0 and θ_1 (black, solid) and its corresponding convex envelope (orange, dashed). (e) Zero-level sets for the parameters $\theta_{0.2}$ and (f) $\theta_{0.8}$.

ellipses (GOCELL) representation of the image. In Eq. (5.5), the pointwise maximum of the family of linear functions $s_1, \ldots, s_m$ is convex in θ, but not affine. Thus, in Eq. (5.5), the composition of the convex loss function $\phi(y(x); \cdot)$, as given by Eq. (5.3), and the multi-object model in Eq. (5.4) is non-convex (cf. Section 2.2.2). An example of this non-convexity is provided in Figure 5.4, which illustrates the energy function along a straight line in the parameter space. The minimization of non-convex energies is generally difficult. In our case, convex envelope-based reformulations (e.g., [118]) are not applicable, because the convex envelope (see Section 2.2.3) of the energy in Eq. (5.5) possesses an infinite number of global minimizers, which are far from being optimal with respect to the energy function (e.g., $\rho = 0.5$ in Figure 5.4d). Also, the sequential approximation scheme from Section 4.2.1 is not applicable, since the multi-object model in Eq. (5.4) is non-polynomial. However, global minimization of the energy in Eq. (5.5) is tractable if the model activity regions are assumed to be unions of adjacent subregions, which we call *region fragments*, as detailed below.

5.3.2 Global energy minimization

In this section, we derive a global minimization scheme for the energy function in Eq. (5.5). Since the loss function $\phi(y(x); \cdot)$ is monotonously decreasing for $y(x) > 0$, the maximization of s_k with respect to $k \in [m]$ is equivalent to the minimization of $\phi(y(x); s_k(x))$ for non-negative $y(x)$. For negative $y(x)$, the minimization is equivalent to the maximization of $\phi(y(x); s_k(x))$, since then $\phi(y(x); \cdot)$ increases

monotonously. Thus, the energy of the multi-object model can be written as

$$\begin{aligned}\psi(\theta) = \sum_{x\in\Omega} \Big([y(x) \geq 0] \cdot \min_{k\in[m]} \left\{ \phi(y(x); s_k(x)) \right\} \Big) \\ + \Big([y(x) < 0] \cdot \max_{k\in[m]} \left\{ \phi(y(x); s_k(x)) \right\} \Big),\end{aligned} \tag{5.6}$$

where [statement] = {1 if statement = true; 0 else} are the Iverson brackets.

The loss function ϕ only attains non-negative values. Thus, the sum over $\phi(y(x); s_k(x))$ for $k \in [m]$ is an upper bound of the maximal $\phi(y(x); s_k(x))$. Using this upper bound to reformulate Eq. (5.6) leads to

$$\begin{aligned}\bar{\psi}(\theta) = \sum_{x\in\Omega} \Big([y(x) \geq 0] \cdot \min_{k\in[m]} \left\{ \phi(y(x); s_k(x)) \right\} \Big) \\ + \Big([y(x) < 0] \cdot \sum_{k\in[m]} \phi(y(x); s_k(x)) \Big)\end{aligned} \tag{5.7}$$

with the property $0 \leq \psi(\theta) \leq \bar{\psi}(\theta)$ for all θ. Hence, the minimization of the upper bound $\bar{\psi}$ also minimizes the energy ψ.

The pointwise minimum of $\phi(y(x); s_k(x))$ over $k \in [m]$, as given in Eq. (5.7), can be written as a pointwise minimization with respect to a binary indicator vector $z(x)$. We hence introduce a binary vector $z(x) \in \{0,1\}^m$ for each image point x, where $z_k(x) = 1$ means that the k-th model is active at x. Then, the energy in Eq. (5.7) can be expressed as a pointwise minimization with respect to z, that is

$$\bar{\psi}(\theta) = \min_{z\in Z} \bar{\psi}_z(\theta) \tag{5.8a}$$

with

$$\begin{aligned}\bar{\psi}_z(\theta) = \sum_{x\in\Omega} \sum_{k\in[m]} [y(x) \geq 0] \cdot z_k(x) \cdot \phi(y(x); s_k(x)) \\ + [y(x) < 0] \cdot \phi(y(x); s_k(x)).\end{aligned} \tag{5.8b}$$

The constraint $z \in Z = \left\{ z : \mathbb{R}^2 \to \{0,1\}^m \middle| \mathbb{1}_m^\top \cdot z(x) \geq 1 \, \forall x \in \Omega \right\}$ and $\mathbb{1}_m^\top$ is an $1 \times m$ vector of values one, enforces that there must be an active model for each $x \in \Omega$. Reordering the two sums in Eq. (5.8b) leads to

$$\bar{\psi}_z(\theta) = \sum_{k\in[m]} \sum_{x\in\Omega} [x \in X_k \vee y(x) < 0] \cdot \phi(y(x); s_k(x)), \tag{5.9}$$

where $X_k = \{x \in \Omega | z_k(x) = 1\}$ is the activity region of the k-th shape model. The minimization of $\bar{\psi}_z(\theta)$ with respect to z and θ also minimizes the upper bound $\bar{\psi}$ of the energy ψ and thus determines the optimal fitting of the multi-object model in Eq. (5.4) to the image data. The energy $\bar{\psi}_z(\theta)$ is convex in θ for fixed z, but it is

non-smooth in z. However, since the order of minimization with respect to z and $\theta = \left(\theta^{(1)}, \ldots, \theta^{(m)}\right)$ is interchangeable, the minimization of Eq. (5.9) boils down to

$$\inf_{\theta} \min_{z \in Z} \bar{\psi}_z(\theta) = \min_{z \in Z} \Big\{ \sum_{k \in [m]} \inf_{\theta^{(k)}} \sum_{x \in \Omega} [x \in X_k \vee y(x) < 0] \cdot \phi\left(y(x); s_k(x)\right) \Big\}. \quad (5.10)$$

Notably, if $X_k = \emptyset$, then $\inf_{\theta^{(k)}} \sum_{x \in \Omega} [x \in X_k \vee y(x) < 0] \cdot \phi\left(y(x); s_k(x)\right) = 0$. We use an indicator vector $d \in \{0,1\}^m$ and define $d_k = 0$ as an equivalent representation of $X_k = \emptyset$. With this substitution ($d_k = [X_k = \emptyset]$), we then obtain

$$\inf_{\theta} \min_{z \in Z} \bar{\psi}_z(\theta) = \min_{z \in Z} \langle d(z), f(z) \rangle, \quad (5.11)$$

as a reformulation of Eq. (5.10), where $c \in \mathbb{R}_+^m$ is a vector with the components

$$c_k = \inf_{\theta^{(k)}} \psi_{\omega_k}\left(\theta^{(k)}\right), \qquad \omega_k = X_k \cup \{x \in \Omega | y(x) < 0\} \quad (5.12)$$

and the convex energy ψ_{ω_k} is given by Eq. (5.3). Below, we describe the solution of Eq. (5.11) by characterizing the model activity regions $X_1, \ldots, X_m$ as a subset of finitely many fixed region prototypes.

Region prototypes

Formally, let $\hat{X}_1, \ldots, \hat{X}_m$ be the model activity regions, which correspond to the optimal $z \in Z$ with respect to Eq. (5.11). Then, each $\hat{X}_k$, which is non-empty ($d_k = 1$), is unique among the regions $\hat{X}_1, \ldots, \hat{X}_m$, because otherwise the optimality assumption is contradicted due to $c_1, \ldots, c_m \geq 0$. Hence, $\hat{X}_1, \ldots, \hat{X}_m$ form a subset of an overcomplete set

$$\mathscr{U} = \{X_1, \ldots, X_n\} \quad (5.13)$$

of $n \geq m$ region prototypes, as detailed in the next paragraph. Using a slightly different connotation of the vector d, each region prototype $X_k \in \mathscr{U}$ is either not included in the solution ($d_k = 0$) or it is included once ($d_k = 1$). Thus, the minimization in Eq. (5.11) can be solved by minimizing with respect to u directly instead of z,

$$\inf_{\theta} \min_{z \in Z} \bar{\psi}_z(\theta) = \min_{d \in D} \langle c, u \rangle, \quad (5.14a)$$

where c and u are now n-dimensional and

$$D = \Big\{ d \in \{0,1\}^n \;\Big|\; \sum_{k \in [n]} [x \in X_k] \cdot d_k \geq 1 \; \forall x \in \Omega \Big\}. \quad (5.14b)$$

The vector c in Eq. (5.14a) is independent of z and d, since it now represents the energies in Eq. (5.12) of the region prototypes $X_1, \ldots, X_n$, which are fixed. The polytope constraint $\sum_{k\in[m]} [x \in X_k] \cdot d_k \geq 1 \, \forall x \in \Omega$ in Eq. (5.14b) enforces that the union of the included prototypes equals Ω, that is, covers the whole image.

The vector c is invariant to the values of $z(x)$ at image points $x \in \Omega$ with $y(x) < 0$, since in Eq. (5.12), the energy ψ_{ω_k} is minimized for the *union* of the region prototype X_k and all image points x with $y(x) < 0$. For the computation of the set $\mathscr{U}$ in Eq. (5.13), it is hence sufficient to consider only those region prototypes $X_1, \ldots, X_n$, which differ with regard to the image foreground (image points x with $y(x) > 0$). Thus, each optimal region $\hat{X}_k$ covers all image points of a *single* cell nucleus and an *arbitrary* part of the image background. This observation motivates two mild assumptions, which prove to be convenient for the characterization of $\mathscr{U}$ as a set of a computationally tractable cardinality:

1. *Model activity regions are simply connected.* This assumption is reasonable since cell nuclei are physically connected objects without holes, although they might *appear* as objects with holes due to staining.

2. *Each model activity region contains one or more local intensity peaks.* This is generally the case, since cell nuclei are brighter than the image background in fluorescence microscopy images. Correspondingly, *the boundary of an activity region is located at an intensity valley.* However, not all intensity valleys represent boundaries of activity regions.

Based on the first assumption, we characterize each region prototype X in $\mathscr{U}$ as a simply connected union of region *fragments* U. Each fragment $u \in U$ is formed around a local intensity peak $r(u)$ and the fragments are separated by intensity valleys, due to the second assumption. We propose Algorithm 5.1 to generate U and the prototype set $\mathscr{U}$. The algorithm constructs the fragment adjacency graph $(U, \mathcal{E})$, where $\mathcal{E} \subseteq U \times U$ and the adjacency of two fragments $u, v \in U$ with sufficiently close local intensity peaks $r(u), r(v)$ is represented by the edge $\{u, v\} \in \mathcal{E}$. The region prototypes $\mathscr{U}$ are then obtained as locally confined subgraphs. The cardinality of the generated set $\mathscr{U}$ is smaller than $\#U \cdot 2^{\Delta^h}$, where Δ is the maximum degree of the fragment adjacency graph $(U, \mathcal{E})$ and h is the maximum search depth. Thus, for any fixed h and degree Δ, the cardinality of $\mathscr{U}$ grows linearly with the number $\#U$ of fragments. The number of fragments is controlled either by choosing the smoothing strength σ of the Gaussian filter or by limiting the set Π in Algorithm 5.1 to a fixed number of most significant local intensity maxima. Although we do not control Δ directly, the average degree of the fragment adjacency graph is strictly smaller than 6, since the graph is planar. The run time complexity of Algorithm 5.1 is at worst quadratic in the number of generated prototypes, since duplicate prototypes must be sorted out to form the prototype set $\mathscr{U}$.

Algorithm 5.1: Generating the fragments U and the prototype set $\mathscr{U}$.

input: Image $g : \Omega \to \mathbb{R}_{\geq 0}$, smoothing strength $\sigma \geq 0$, relative intensity threshold $0 \leq \varepsilon \leq 1$, maximum search depth $h \geq 0$, minimum seed distance $\delta_{\text{seed}} \geq 1$, maximum fragments distance $\delta_{\text{frag}} \geq 0$.

Apply Gaussian filter with standard deviation σ to the image g;
Let $B(\hat{x})$ be $\{x \in \Omega | \|\hat{x} - x\| \leq \delta_{\text{seed}}\}$;
$\Pi \leftarrow \left\{x \in \Omega \middle| g(x) = \max_{x' \in B(x)} g(x') \wedge (1-\varepsilon) \cdot g(x) \geq \min_{x' \in B(x)} g(x')\right\}$;
$U \leftarrow \left\{u \subseteq \Omega \middle| u \text{ is a region of } \Pi\text{-seeded watershed transform of } g\right\}$;
Let $r(u)$ be the centroid of $\Pi \cap u$;
$\mathcal{E} \leftarrow \left\{\{u, v\} \in U \times U \middle| u \text{ is adjacent to } v \wedge \|r(u) - r(v)\| \leq \delta_{\text{frag}}\right\}$;
$\mathscr{U} \leftarrow \emptyset$;
for $u \in U$ **do**
  for *each simply connected subgraph* $(U', \mathcal{E}')$ *of* $(U, \mathcal{E})$ *induced by* $U' \subseteq U$ *with* $u \in U'$ *and* $\max_{v \in U'} \text{dist}_{\mathcal{E}'}(u, v) \leq h$ **do**
    $\mathscr{U} \leftarrow \mathscr{U} \cup \{X\}$ where $X = \bigcup_{v \in U'} v$;
return $U, \mathscr{U}$;

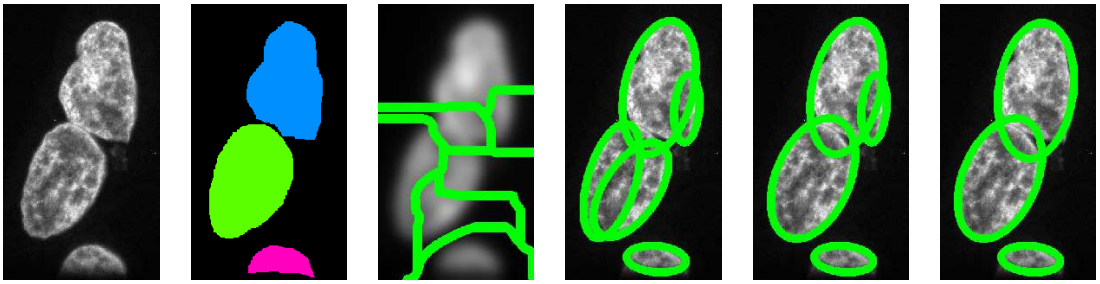
Incorporation of region fragments enables formulating the minimization in Eq. (5.14a) as an integer linear program (ILP):

$$\begin{aligned} &\text{minimize} && \langle c, d \rangle \\ &\text{subject to} && \sum_{k \in [n]} [u \subseteq X_k] \cdot d_k \geq 1 \quad \forall u \in U, \\ & && d \in \{0,1\}^n. \end{aligned} \tag{5.15}$$

Since the region prototypes $X_1, \ldots, X_n$ are independent of d, the computation of each component of the vector c amounts to solving an individual convex program, as detailed in Section 5.3.3 below. The combinatorial minimization by the ILP in Eq. (5.15) yields the vector d. This determines, which of the n prototypes are to be used to form the model activity regions ($d_k = 1$), subject to the constraint that an active model exists at each point of the image. Thus, our derivation shows that model fitting using the multi-object model in Eq. (5.4) corresponds to solving the **NP**-hard *min-weight set-cover* problem (see Definition 2.2) in Eq. (5.15).

Tolerance for shape irregularities

Due to the overcompleteness of the prototype set $\mathscr{U}$, the number m of models being fitted by the ILP in Eq. (5.15) only has the upper bound $n \geq m$, which is the cardinality of the prototype set $\mathscr{U}$. Since this number is usually larger than the number of cell nuclei in an image, this likely leads to oversegmentation of non-ideally elliptical cell nuclei, as the example in Figure 5.5 shows. The two large, slightly irregularly shaped cell nuclei in Figure 5.5a are falsely split (Figure 5.5d).



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

Figure 5.5. Influence of the tolerance for shape irregularities β in Eq. (5.16) on the segmentation result. (a) Original image section (HeLa cells). (b) Ground truth. (c) Model activity region fragments U (green contour) generated from intensity peaks of the smoothed image (Algorithm 5.1). (d) Segmentation result (green contour) using $\beta = 0$, (e) $\beta = 0.6 \cdot c_{\mathrm{m}}$, and (f) $\beta = 1 \cdot c_{\mathrm{m}}$, where c_{m} is the median of the components of the vector c.

The reason is that, for our multi-object model, these cell nuclei are rather cases of two overlapping cell nuclei, than individual objects. To cope with that, we incorporate a tolerance for slight shape irregularities by demanding additional sparsity for the solution d in Eq. (5.15). This is done by introducing a penalty $\beta > 0$ for each selected prototype ($d_k = 1$) into the objective function, which yields

$$\begin{aligned} &\text{minimize} && \langle c, d \rangle + \beta \cdot \langle \mathbb{1}_n, d \rangle \\ &\text{subject to} && \sum_{k \in [n]} [u \subseteq X_k] \cdot d_k \geq 1 \quad \forall u \in U, \\ & && d \in \{0,1\}^n . \end{aligned} \tag{5.16}$$

The ILP in Eq. (5.16) is identical to Eq. (5.15) for $\beta = 0$. It is beneficial to choose a value of β in the same range as the components of the vector c. In our experiments, we specified β as a multiple of the median c_{m} of the components of the vector c. Figure 5.5e and Figure 5.5f show that the segmentation result improves by increasing β up to $\beta = c_{\mathrm{m}}$. Choosing a too large value for β might cause false merges of closely located cell nuclei.

Joint cell segmentation and cluster splitting by combinatorial optimization

The ILP in Eq. (5.16) incorporates both shape and intensity information through the components of the vector c defined in Eq. (5.12). Solving this ILP boils down to merging adjacent region fragments if this improves the value of the objective function, as described in Section 5.3.3 below. This enables coping not only with touching and partially overlapping cell nuclei, but also with intensity inhomogeneities of the cell nucleus due to staining.

5.3.3 Optimization methods and implementation details

To compute the components of the vector c according to Eq. (5.12), the energy in Eq. (5.3) needs to be minimized for each region prototype. This energy depends on the intensity offset τ of the intensity model y defined in Eq. (4.2). In fluorescence microscopy images, we need to cope with cell nuclei of varying intensities and non-homogeneous image backgrounds. Thus, instead of using a global intensity offset τ for the whole image, we computed τ adaptively for each region prototype. The value of τ was selected either using Otsu thresholding [33] or by determining the first mode in the intensity distribution of the image region, obtained by kernel density estimation [119] using Gaussian kernels.

In order to accelerate the computation of the vector c, we exploited that the vector components are independent of each other by computing the components in parallel. For each component, we solved the minimization of the energy in Eq. (5.3) using the iterative second-order solver [120] for unconstrained convex optimization (see Section 2.2.5), which exploits the positive definiteness of the Hessian matrix $\nabla^2\psi_\omega$ for rapid convergence. The first- and second-order derivatives of the energy in Eq. (5.3) are

$$\begin{aligned} \nabla\psi_\omega(\theta) &= -\sum_{x\in\omega} y(x)\cdot\kappa_{x,\theta}\cdot\nabla s(x), \quad \kappa_{x,\theta} = \frac{1}{1+\exp\left(y(x)\cdot s(x;\theta)\right)}, \\ \nabla^2\psi_\omega(\theta) &= \sum_{x\in\omega} y(x)^2\cdot\left(\kappa_{x,\theta}-\kappa_{x,\theta}^2\right)\cdot\nabla s(x)\cdot\nabla^\top s(x), \end{aligned} \tag{5.17}$$

where ∇s is the gradient of the linear model defined in Eq. (5.1) with respect to its parameters θ. Choosing the basis $E_1 = \left[\begin{smallmatrix}1&0\\0&0\end{smallmatrix}\right]$, $E_2 = \left[\begin{smallmatrix}0&0\\0&1\end{smallmatrix}\right]$, $E_3 = \left[\begin{smallmatrix}0&1\\1&0\end{smallmatrix}\right]$ to represent the matrix $A = \sum_{i=1}^{3} a_i\cdot E_i$ and $\theta = (a_1, a_2, a_3, b_1, b_2, c)$ for the vectorial representation of the parameters θ, we obtain the gradient of the linear model as $\nabla^\top s(x) = \left(x_1^2, x_2^2, 2x_1x_2, x_1, x_2, 1\right)$. We used the zero-vector for initialization. However, the initialization can be arbitrary due to convexity.

The min-weight set-cover problem in Eq. (5.16) is **NP**-hard. In general, it thus cannot be expected that an exact solution for d is obtained in polynomial time. However, a greedy heuristic [27] is known to determine an approximate solution within an approximation guarantee of factor $H_{\#U}$ or better, where $H_t = \sum_{i=1}^{t} 1/i$ is the t-th harmonic number (see Section 2.2.6). We used the Algorithm 5.2 to solve the ILP in Eq. (5.16), where the greedy heuristic is combined with a local search. The local search merges adjacent regions if this decreases the energy $\tilde{\psi} = \langle d, c + \mathbb{1}_n\cdot\beta\rangle$ of the solution. Hence, the approximation ratio of Algorithm 5.2 is *never worse* than $H_{\#U}$. This conservative lower bound can be tightened *a posteriori* by solving the linear programming (LP) relaxation. Let $\hat{\psi}_{\mathrm{LP}}$ be the exact solution (energy) of the LP relaxation of the ILP in Eq. (5.16). Since $\hat{\psi}_{\mathrm{LP}}$ is a lower bound of the unknown optimal solution $\hat{\psi}$ of Eq. (5.16), the achieved approximation ratio of the solution $\tilde{\psi}$ is at least $\hat{\psi}_{\mathrm{LP}}/\tilde{\psi}$. We found that Algorithm 5.2 succeeded in determining a de-facto

Algorithm 5.2: Determining the global solution of the ILP in Eq. (5.16) within an approximation guarantee of $H_{\#U}$ or better.

input: Vector c, sparsity β, region fragments U, region prototypes $\mathscr{U}$.

Initialize $d \leftarrow 0 \cdot \mathbb{1}_n$; $V \leftarrow U$; $\mathscr{V} \leftarrow \mathscr{U}$;
while $V \neq \emptyset$ **do** *// greedy*
  Set $T_k \leftarrow \frac{c_k+\beta}{|X_k \cap V|}$ for all $k = 1, \ldots, n$ where $\mathscr{U} = \{X_1, \ldots, X_n\}$;
  Set $\hat{k} \in [n]$ so that $T_{\hat{k}} = \min_{k \in [n]} T_k$;
  Set $d_{\hat{k}} \leftarrow 1$; and $V \leftarrow V \setminus \left\{X_{\hat{k}}\right\}$;
while $\mathscr{V} \neq \emptyset$ **do** *// local search*
  Set $\hat{k} \in [n]$ so that $X_{\hat{k}} \in \mathscr{V}$ and $c_{\hat{k}} = \min\{c_k | k \in [n], X_k \in \mathscr{V}\}$;
  if $d_{\hat{k}} = 0 \wedge \exists d' \in \{0,1\}^{\#\mathscr{U}} : d' \leq d \wedge \bigcup_{k \in [n]: d'_k = 1} X_k = X_{\hat{k}}$ **then**
    if $c_{\hat{k}} + \beta < \langle d', f + \beta \cdot \mathbb{1}_n \rangle$ **then**
      Update $d \leftarrow d - d'$; and $d_{\hat{k}} \leftarrow 1$;
  $\mathscr{V} \leftarrow \mathscr{V} \setminus \left\{X_{\hat{k}}\right\}$;
return d;

exact solution ($\hat{\psi}_{\mathrm{LP}}/\tilde{\psi} \geq 99\,\%$) in at least 91.5 % of our experiments (Section 5.4) and the obtained worst lower bound of the ratio was 88.7 %. The run time complexity of Algorithm 5.2 is at worst quadratic in the cardinality of $\mathscr{U}$.

The final segmentation result is given by the subset of the shape models $C_s(\theta^{(1)}), \ldots, C_s(\theta^{(n)})$ which is identified by $d_k = 1$, as determined by Algorithm 5.2. Ellipses with a significant overlap (larger than 40 % or 50 %, depending on the image data) were considered as single objects.

5.4 Evaluation

We have applied our multi-object model-based approach GOCELL (globally optimal collaborating ellipses) to 2-D fluorescence microscopy image data. Our experiments comprise image datasets of five different cell types. We studied the segmentation accuracy as well as the computation time, and performed a comparison with previous methods. To quantify the segmentation accuracy, we used region-based (Dice, Rand, SEG) and contour-based measures (HSD, NSD), which are described in Section 3.4.

5.4.1 Computational complexity

First, we studied the computational complexity of our approach using an example microscopy image of DAPI-stained HeLa cell nuclei (Figure 5.6a). The size of the image is 741×1000 pixels. As described in Section 5.3, the computational complexity of our approach crucially depends on the cardinality of the prototype set $\mathscr{U}$. Therefore, we varied the smoothing strength σ of the Gaussian filter (within

the range [2, 50]) when applying Algorithm 5.1 (using $\varepsilon = 1\,\%$, $h = 2$, $\delta_{\text{seed}} = 20$, $\delta_{\text{frag}} = 100$). For smaller values of σ (little smoothing), more activity region fragments occur. Assuming that the maximum number of fragments adjacent to any single fragment in U (i.e. the maximum degree of the fragment adjacency graph $(U, \mathcal{E})$) is constant, the number of prototypes scales linearly with the number of the fragments. However, when the smoothing strength is lowered, the fragments become more irregularly shaped, and thus the maximum degree of the fragment adjacency graph grows. Hence, the cardinality of the prototype set increases somewhat faster with increasing number of fragments (Figure 5.6b). This is tolerable, since the overall run time grows almost linearly with the number of prototypes (Figure 5.6c), where $\beta = 2 \cdot c_{\text{m}}$ was used for Algorithm 5.2 and τ was determined by Otsu thresholding. Although the computational complexity of Algorithm 5.1 and Algorithm 5.2 is at worst quadratic in the number of the prototypes, they terminate rapidly, since both algorithms consist of only few instructions per iteration. Thus, the overall run time is dominated by the computation of the energy values of the region prototypes and grows linearly with the number of the prototypes (controlled by σ). For each value of σ, eight prototypes were processed in parallel using a regular consumer CPU (Intel(R) Core(TM) i7 860 2.80GHz). The overall run time performance, as a function of the number of region fragments, is shown in Figure 5.6d. Our approach terminated after 1 minute for 16 fragments (using $\sigma = 16$) and after 9.7 minutes for 41 fragments (using $\sigma = 6$). In both cases, a Dice value of 94.6 % was achieved. For less than 16 fragments (corresponding to $\sigma > 16$), the fragments become too coarse and the segmentation accuracy reduces (cf. Dice value for less than 1 minute in Figure 5.6e). For more than 16 fragments (corresponding to $\sigma < 16$), the run time increases but the segmentation accuracy remains high (cf. Dice value for more than 1 minute in Figure 5.6e). Thus, for this example image, $\sigma = 16$ is an optimal trade-off between segmentation accuracy and run time.

For comparison, we also applied another globally optimal approach for cell nuclei segmentation [64]. This approach is also based on a parameterized shape model, but uses a marked point process, that is embedded into a simulated annealing scheme. We applied this approach using circular (MPP) or elliptical (MPPELL) shape models. MPP converged after 38.2 minutes, achieving a Dice value of 52.9 % (Figure 5.6g). MPPELL converged after 35.7 hours with an improved Dice value (61.6 %, Figure 5.6h). In comparison, our approach (GOCELL) yielded a better Dice value of 94.6 % (Figure 5.6i) after only 1 minute. We note that the computation time of our approach can be straightforwardly reduced by parallelization (e.g., using more than eight CPU threads as in our case). This is possible since computing the energy values of the region prototypes (which dominates the run time) can be performed independently from each other. Typically, we obtained a few hundred prototype regions in our experiments (553 or less in 95 % of the images). The ratio

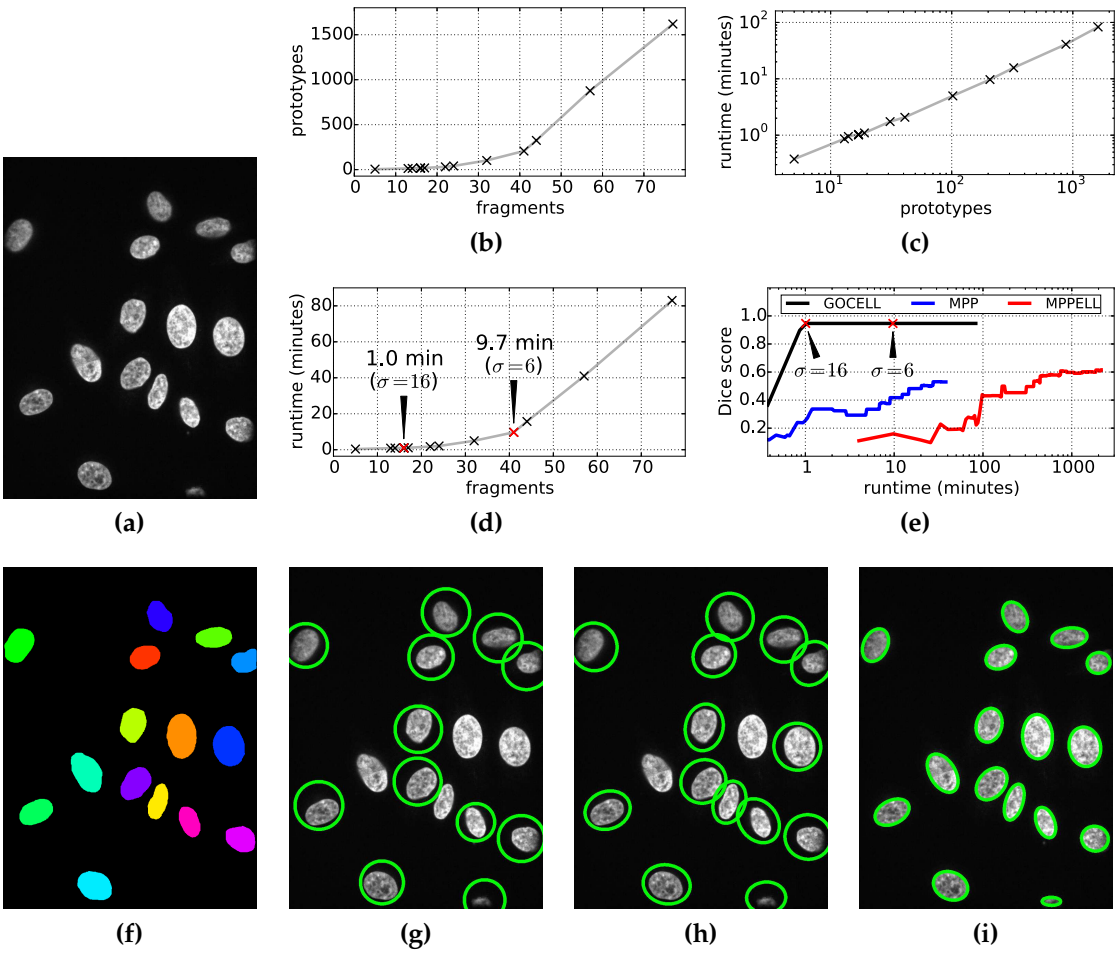


Figure 5.6. Run time performance for an example image. (a) Original image (HeLa cells). (b) Cardinality of the prototype set as a function of the number of activity region fragments. (c) Run time as a function of the cardinality of the prototype set. (d) Run time as a function of the number of activity region fragments. (e) Dice score as a function of the run time of our approach (GOCELL) and an approach based on marked point processes using circular (MPP) or elliptical models (MPPELL). (f) Ground truth segmentation. (g) Segmentation result (green contour) for MPP, (h) MPPELL, (i) and GOCELL.

between the number of region prototypes and the final number of segmented cell nuclei in an image was between 4 and 8 in most cases, the median was 5.4.

5.4.2 Macrophage, HeLa, and Fibroblast datasets

Next, we studied the segmentation performance of our GOCELL approach using DAPI-stained images of three different cell types, corresponding to three different datasets (Macrophage, HeLa, Fibroblast). The datasets are described in Section 3.2.

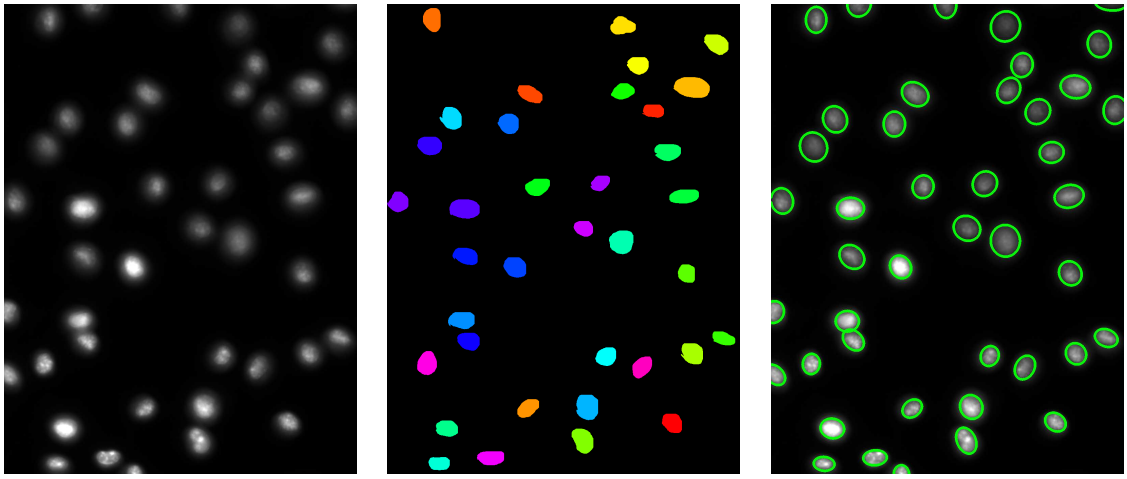
We applied our approach to all three datasets and performed a comparison with standard approaches, including Otsu, Blob-LS, and Blob-RW (described in Section 3.3). For all three methods, we applied pre-processing by Gaussian filtering and post-processing by morphological closing. We optimized the parameters of the three methods as well as their respective pre/post-processing steps individually for each dataset. This was accomplished by an automatic grid search scheme, which maximizes the average Dice and SEG values using two randomly chosen images from each dataset. In contrast, for our GOCELL approach, we did not adapt the parameters individually for each dataset but used the same set of parameters for all three datasets (as described in Section 5.4.1, using $\sigma = 11$).

Macrophage dataset

The Macrophage dataset is difficult due to partially strong image blur (see Figure 5.7). The quantified segmentation results are provided in Table 5.1. It turns out that our approach yields the best result for all three performance measures (SEG, Dice, NSD). Compared to the second-best method (Blob-LS), SEG is improved by 3.5 %P and NSD by 27 %, while for Dice we have a small degradation of 0.7 %P. The highest improvement for SEG is obtained compared to Otsu (6 %P). For this dataset, we also computed results for our approach (GOCELL) when adapting the parameters (as for the other three methods) by reducing the tolerance for shape irregularities to $\beta = 0.6 \cdot c_{\mathrm{m}}$ and using kernel density estimation instead of Otsu thresholding to determine the intensity threshold τ (GOCELL*). This improved the SEG value by 2.5 %P, the Dice value by 1.4 %P, and the NSD value by 23 %.

HeLa dataset

For the HeLa dataset, our approach (GOCELL) performed better than all other methods. Using dataset-specific adaptations by increasing the tolerance for shape irregularities ($\beta = 3c_{\mathrm{m}}$) further improved SEG by 1.1 %P and NSD by 19 % (GOCELL*). The other parameters remained the same as in Section 5.4.1.

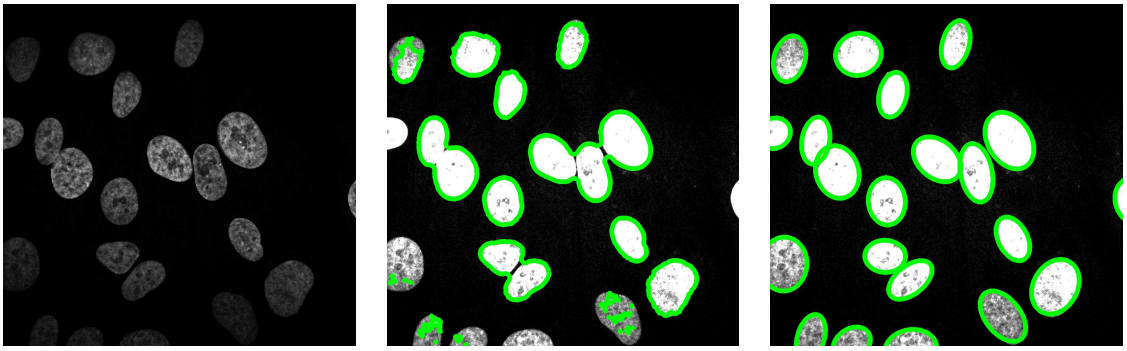


(a) (b) (c)

Figure 5.7. Example segmentation results for the Macrophage dataset. (a) Original image. (b) Ground truth segmentation. (c) Segmentation result (green contour) using our GOCELL approach.

	SEG	Dice	NSD	Parameter sets
Macrophages				
Otsu	66.7 %	81.3 %	0.268	Adapted
Blob-LS	69.2 %	81.4 %	0.227	Adapted
Blob-RW	67.8 %	79.0 %	0.176	Adapted
GOCELL	72.7 %	80.7 %	0.166	Same for all
GOCELL*	**75.2 %**	**82.1 %**	**0.127**	Adapted
HeLa cells				
Otsu	85.4 %	93.7 %	0.077	Adapted
Blob-LS	85.4 %	93.2 %	0.063	Adapted
Blob-RW	68.3 %	81.3 %	0.146	Adapted
GOCELL	87.9 %	**94.3 %**	0.037	Same for all
GOCELL*	**89.0 %**	**94.3 %**	**0.030**	Adapted
Fibroblasts				
Manual	92.3 %	89.5 %	0.008	
Otsu	78.3 %	86.4 %	0.135	Adapted
Blob-LS	71.5 %	83.4 %	0.178	Adapted
Blob-RW	29.3 %	63.8 %	0.281	Adapted
GOCELL	**93.1 %**	**90.9 %**	**0.012**	Same for all

Table 5.1. Segmentation performance of our approach using the same parameter configuration for all three datasets (GOCELL) or using dataset-specific parameter configurations (GOCELL*) compared to manual segmentation and standard approaches, which were optimized for each dataset. The best results for each dataset are highlighted in bold.



(a) (b) (c)

Figure 5.8. Example segmentation results for the Fibroblast dataset. (a) Original image. (b) Segmentation result (green contour) using Otsu thresholding. (c) Segmentation result (green contour) using our GOCELL approach.

Fibroblast dataset

For the Fibroblast dataset, our approach (GOCELL) performed better than the other approaches for all three performance measures. An example of a segmentation result is shown in Figure 5.8. It can be seen that our approach effectively separates touching cell nuclei since shape information is exploited. In contrast, Otsu thresholding, which performed second-best on this dataset, falsely merges closely located cell nuclei (Figure 5.8b). Blob-LS and Blob-RW performed worse since their initialization is prone to the densely located and non-elliptical cell nuclei. For a comparison with the performance of manual segmentation, a human expert manually segmented 34 images, yielding a SEG value of 92.3 % and Dice value of 89.5 %. Thus for our approach it turns out that the SEG and Dice values are higher compared to manual segmentation.

Conclusion

Overall, our approach performed best for all three datasets. The best results were obtained using dataset-specific adaptations for our approach (GOCELL*). However, more importantly, using fixed parameters for all three datasets for our approach (GOCELL) yielded better or comparable results than the other approaches, despite of the heterogeneity of the datasets (cf. Figure 5.7 and Figure 5.8).

5.4.3 NIH3T3 dataset

We also applied our approach to the publicly available NIH3T3 dataset (described in Section 3.2). Figure 5.9 shows an example image with corresponding ground truth data and segmentation results. For our approach, we used $\sigma = 6$ and $\beta = 0.3 \cdot c_{\mathrm{m}}$ for all images of the dataset, the other parameters remained the same as in Section 5.4.1. To better cope with the strong background intensity inhomogeneities in this dataset,

	SEG	Dice	Rand	HSD	NSD	Merged	Split
CVX-LS	65.2 %	85.3 %	90.5 %	14.2	0.12	1.6	**0.0**
TM	—	—	88 %	134.1	0.29	4.0	**0.0**
Blob-GC	—	—	91.5 %	10.3	—	1.6	2.4
RPL	—	90.6 %	93.2 %	14.1	0.09	—	—
Bayes-LS	—	86.3 %	—	14.3	—	—	—
SEF-SELF	79.7 %	88.7 %	91.8 %	9.5	0.08	0.8	1.3
CVXELL	74.5 %	87.4 %	90.6 %	14.3	0.14	1.4	0.3
GOCELL	**83.7 %**	**91.5 %**	**93.6 %**	**8.3**	**0.06**	**0.7**	0.4

Table 5.2. Segmentation performance of our GOCELL approach for the NIH3T3 dataset compared to previous approaches. Not available results are indicated by "—". The best results are highlighted in bold.

we employed local background subtraction based on the minimal intensities of the Gaussian-filtered image (standard deviation 1) within circular neighborhoods (50 pixels radius) of each pixel (see Figure 5.10). In addition, segmented objects with a radius smaller than 22 pixels were discarded to eliminate the visible debris objects. We quantitatively compared the segmentation performance of our GOCELL approach to our previous CVXELL approach which is based on a single-object model (Chapter 4), as well as six state-of-the-art methods including CVX-LS [51], TM [110], Blob-GC [77], RPL[112], Bayes-LS [52], and SEG-SELF [65] (described in Section 3.3).

The results of the different approaches are given in Table 5.2. The performance values for TM, Blob-GC, RPL, and Bayes-LS were reported in publications by the authors. It turns out that our approach (GOCELL) achieved the best results for all region- (SEG, Dice, Rand) and contour-based (HSD, NSD) measures. As additional object-based performance measures we determined the average numbers of falsely merged / split cell nuclei per image (described in Section 3.4). Our GOCELL approach yielded the lowest number of falsely merged cell nuclei and the second-lowest number of falsely split cell nuclei. Although SEG-SELF yielded only slightly more falsely merged cell nuclei per image (0.8 compared to 0.7), much more falsely split cell nuclei were obtained (1.3 compared to 0.4). Also, the overall performance of SEG-SELF was worse. The lowest number of falsely split cell nuclei was achieved by CVX-LS and TM, but these approaches performed worse with regard to all other performance measures. In particular, falsely merged cell nuclei occurred more than twice as often as for our approach. Compared to RPL, our approach yields a slight improvement for the region-based measures (Dice improved by 0.9 %P, Rand by 0.4 %P, SEG was not reported for RPL), but a significant improvement for the contour-based measures (HSD improved by 41 %, NSD by 33 %). Considering all performance measures, our approach performed overall best on this dataset.

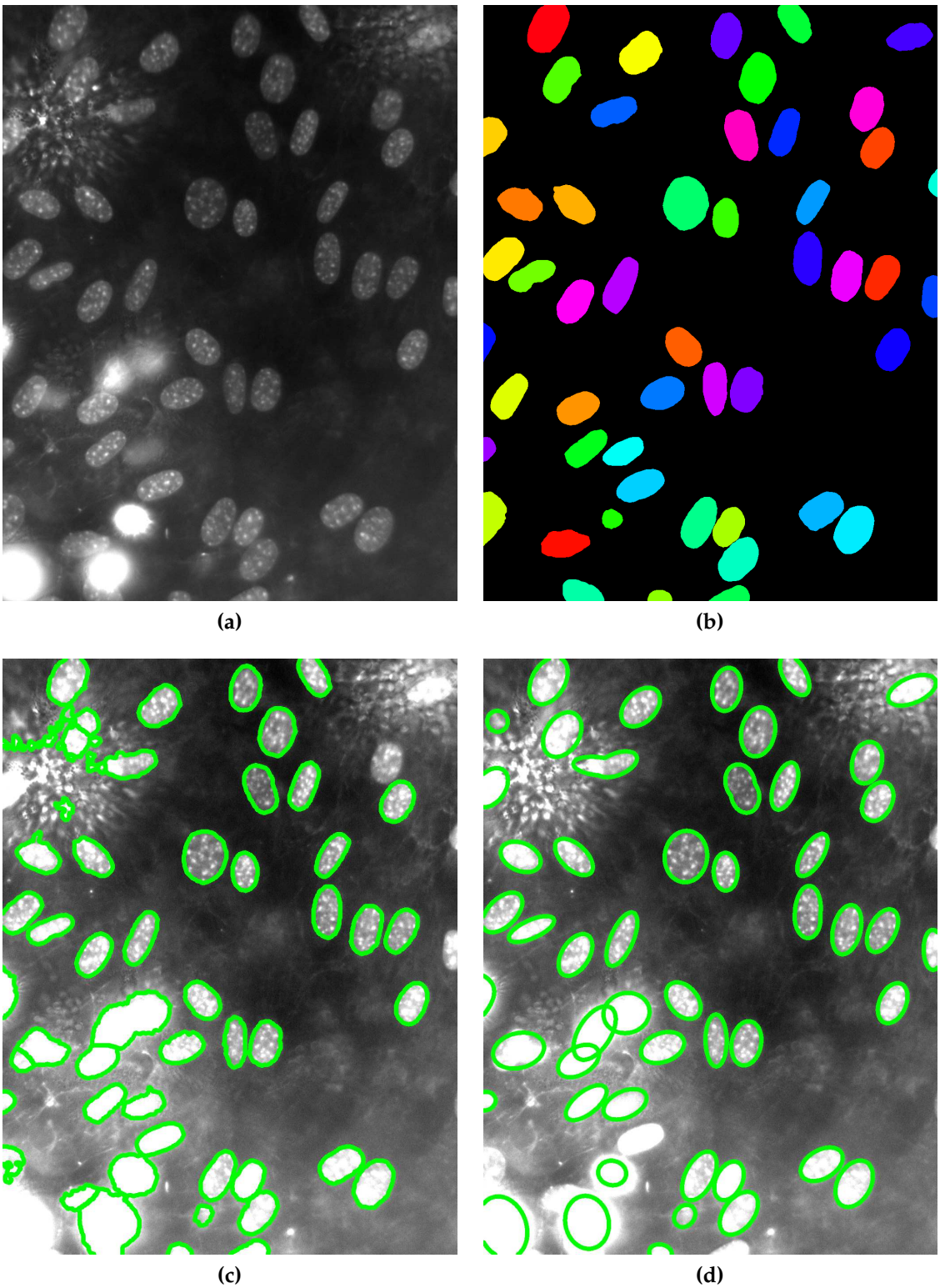
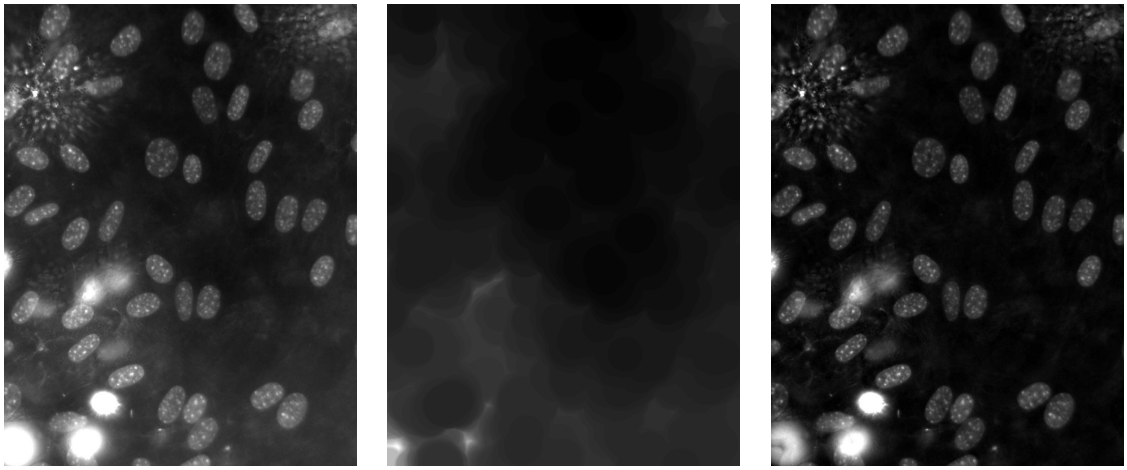


Figure 5.9. Example segmentation results for the NIH3T3 dataset. (a) Original image (contrast-enhanced). (b) Ground truth segmentation. (c) Segmentation result (green contour) using the SEG-SELF approach. (d) Segmentation result (green contour) using our GOCELL approach.



(a) (b) (c)

Figure 5.10. Example for preprocessing of the NIH3T3 dataset. (a) Original image. (b) Computed background. (c) Results after local background subtraction.

5.4.4 GOWT1 datasets

We also applied our approach to two image sets of mouse embryonic stem cells (GOWT1) from the IEEE ISBI Cell Tracking Challenge training data [104]. The two image sets are temporal image sequences with fully annotated segmentation ground truth for eight images and partially annotated ground truth for the other images (see Section 3.2 for details). Since for the partially annotated images ground truth is not available for all objects, using a performance measure which is not invariant to false-positive detections would yield misleading results. In previous work (e.g., [80]), only SEG was used as performance measure for the whole dataset, since it is invariant to false-positive detections and reflects the object-based segmentation performance. In our evaluation, we also used SEG for the whole dataset, but additionally used Dice for the fully-annotated images of the dataset.

The GOWT1 datasets are challenging due to a partially low signal-to-noise ratio, the visible presence of the cell nucleoli (distinct dark regions within individual cell nuclei), and since for many images only the difficult cell nuclei were annotated in the ground truth. An example image from each dataset, the corresponding ground truth, and the segmentation result of our GOCELL approach are shown in Figure 5.11. Due to the nucleoli, the cell nuclei often appear rather as bright rings than as ellipses. Therefore, we pre-processed the images using a Laplacian of Gaussian filter to detect small dark regions and decreased the contrast based on the mean intensities inside and outside these regions. For both datasets, we used $\sigma = 10$ and $h = 1$. We used kernel density estimation to determine the intensity offset τ. To reliably separate very noisy, but almost ideally elliptical nuclei (e.g., Figure 5.11, first row, bottom-right), we reduced the tolerance for shape irregularities to $\beta = 0.1 \cdot c_{\mathrm{m}}$

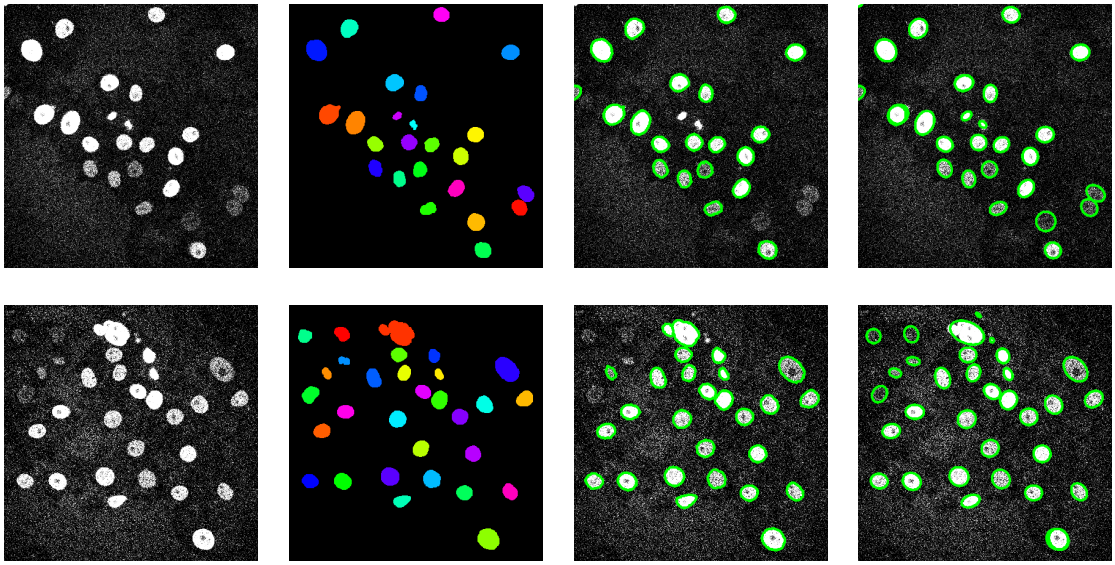


Figure 5.11. Example segmentation results for the GOWT1 dataset 1 (first row) and GOWT1 dataset 2 (second row). (a) Original images (contrast-enhanced). (b) Ground truth segmentations. (c) Segmentation results (green contour) using the SEG-SELF approach. (d) Segmentation results (green contour) using our GOCELL approach.

and $\beta = 0$ for GOWT1 dataset 1 and 2, respectively. All other parameters remained the same as in Section 5.4.1.

We compared the performance of our approach on the two GOWT1 datasets to our previous CVXELL approach (Chapter 4) and four state-of-the-art methods including KTH [111], Blob-WS [79], CPN [80], and SEG-SELF [65] (described in Section 3.3). Since both Blob-WS and CPN rely on temporal information to determine the final segmentation result, they are not applicable to individual images.

The results for all approaches are given in Table 5.3. The performance values for KTH, Blob-WS, and CPN were provided in Akram et al. [80]. For GOWT1 dataset 1 and the SEG measure, GOCELL performed not only better than BLOB (+10.3 %P) and SEG-SELF (+32.4 %P), but also significantly better than KTH (+16 %P), which achieved the best overall result for segmentation in the ISBI challenge [104]. GOCELL yielded a slightly worse result compared to CPN (SEG −0.6 %P), which, however, exploits temporal information.

For GOWT1 dataset 2, our approach (GOCELL) achieved a slightly lower SEG value (−0.3 %P) than our previous CVXELL approach, but a significantly better Dice value (+5.1 %P). More importantly, our approach outperformed SEG-SELF by 9.2 %P, KTH by 1.6 %P, and the tracking-based approaches CPN and Blob-WS by 3.7 %P and 0.5 %P, respectively. Thus, GOCELL performed overall best on this dataset.

	Tracking-based		Single-image segmentation			
	Blob-WS	CPN	KTH	SEG-SELF	CVXELL	GOCELL
GOWT1 dataset 1						
SEG	74.2 %	**85.1 %**	68.5 %	52.1 %	82.1 %	84.5 %
Dice	—	—	—	88.7 %	63.7 %	**94.0 %**
GOWT1 dataset 2						
SEG	90.5 %	87.3 %	89.4 %	81.8 %	**91.3 %**	91.0 %
Dice	—	—	—	91.8 %	89.4 %	**94.5 %**

Table 5.3. Segmentation performance of our GOCELL approach for the GOWT1 datasets compared to previous approaches. The tracking-based approaches (Blob-WS and CPN) exploit information from several images (temporal coherence) in contrast to the other approaches, which use only the information of a single image. The Dice measure was computed for those images only, for which fully labeled ground truth was available (four images per dataset). Not available results are indicated by "—". The best results are highlighted in bold.

5.5 Discussion

We have introduced a new globally optimal approach for cell nuclei segmentation in fluorescence microscopy images. The approach is based on implicitly parameterized elliptical shape models and incorporates intensity information by a contrast-invariant energy function. An advantage of the single-object model is that the corresponding energy is convex. This means that the energy can be directly globally minimized using an arbitrary initialization. However, since this model represents a single object, prior extraction of image regions is required, which contain at most one cell nucleus. To perform segmentation, which is globally optimal with respect to the *entire* image, we generalized the model so that multiple shape models collaboratively represent *all* objects of an image. The corresponding energy function is non-convex and global minimization is challenging. However, we have derived a global minimization scheme, which is based on activity regions of the individual shape models. Our theoretical considerations have shown that the global solution is invariant with respect to specific non-identical regions. We exploited this observation to reduce the set of possible model activity regions to a computationally tractable size using an overcomplete set of region prototypes. Each region prototype is associated with a non-negative energy, which is the infimum of a convex function. We showed, that the non-convex multi-object model energy is minimized by choosing an energy-minimal subset of region prototypes, which covers the whole image. Since computing this *min-weight set-cover* is **NP**-hard, a fast approximation algorithm has been used, which is *guaranteed* to determine a solution close to global optimality. In addition, global optimality was checked *a*

posteriori and we found, that the global solution was *exactly* determined in at least 91.5 % of our experiments comprising 380 images.

Previous approaches, which jointly exploit shape and intensity information (e.g., [64]), assemble segmentation results by selecting object segmentation masks. In contrast, the model activity regions used in our approach only coarsely subdivide an image compared to the final segmentation result. Hence, the solution space for determining the optimal subset of region prototypes is smaller than if considering directly the segmented objects as in previous approaches. This is advantageous, since combinatorial optimization is computationally challenging. Our combinatorial formulation is also fundamentally different on the conceptual level. Previous approaches [63, 64, 78] identified favorable segmentation candidates by *negative* energy values while performing energy *minimization*. Hence, object interaction models (mutual exclusion constraints) were required to prevent non-meaningful solutions, such as the trivial solution (selection of all candidates with negative energy values). Since, however, the energies in our approach are *non-negative*, mutual exclusion constraints are not required, and an object interaction model (e.g., maximum allowed object overlap) is not needed. Our non-negative energy minimization scheme intrinsically favors sparse solutions and enables our approach to naturally cope with touching and overlapping cell nuclei. To better cope with non-elliptical cell nuclei, we included a parameter β in our energy minimization scheme, which controls the tendency of recognizing such cell nuclei as single objects.

The computational complexity of our approach depends on the number of the region prototypes. We used *region fragments* to approximate the set of all permissible region prototypes by a set of computationally tractable cardinality. By controlling the coarseness of the fragments, the error introduced by the approximation is balanced against the computation time. Ideally, the fragments are as coarse as possible, but no fragment should cover more than one cell nucleus. Thus, the choice is intuitive and can be adapted in advance. The run time is dominated by the computation of the energies of the individual region prototypes, which can be highly reduced by parallelization. In addition, adaptation of the parameter β does not require recomputing these energies and is thus fast. Hence, our approach is suitable for high-throughput applications and large datasets.

We applied our approach to fluorescence microscopy images of five different cell types and performed a quantitative comparison with previous methods. We demonstrated the robustness of our approach for datasets of three different cell types (macrophages, HeLa cells, and fibroblasts), achieving equally good or improved results using a fixed set of parameters compared to standard approaches using individually optimized parameters for each of the three datasets. For the NIH3T3 benchmark dataset [103], our approach performed best, achieving a relatively low number of falsely merged/split cell nuclei compared to previous approaches. This highlights the effectiveness of our approach which performs joint

segmentation and cluster splitting, as opposed to explicit cluster splitting (e.g., [52]). Our approach exploits both shape and intensity information jointly, while in [65] the image intensities were not directly exploited for cluster splitting. In our approach, elliptical models are fitted directly to the image intensities. For the two GOWT1 datasets [104], our approach achieved competitive or improved results compared to state-of-the-art methods, including two tracking-based approaches which exploit the temporal coherence of the datasets. Moreover, our approach performed overall best among those methods which do not exploit temporal information and are applicable to individual images.

Chapter 6

Segmentation using superadditivity and deformable shape models

6.1 Introduction

Parametric and non-parametric deformable models, also known as active contours, have a long-lasting history in cell segmentation and computer vision in general (see Section 2.3.2). When equipped with a shape prior, such models can be denoted as *deformable shape models (DSMs)*. Parametric shape models, which are not active contours but capable of performing local deformations, can be considered as DSMs, too. In general, DSMs can be *non-parametric* or *parametric*:

1. **Non-parametric DSMs.** Non-parametric DSMs are based on the variational level set framework and shape regularization (e.g., [56, 57, 58, 59, 60, 61]). Priors used for shape regularization include elliptical shape priors [56], distance-based priors [61], statistical shape priors [57, 59, 60], and star-shape priors [58].

2. **Parametric DSMs.** In contrast to those parametric models which are limited to circular (e.g, [41, 67]) or elliptical (e.g., [42, 62, 63, 64, 65, 66]) shapes, parametric DSMs (e.g., [43, 44, 68, 69]) are more general and intrinsically cope with non-elliptically shaped cell nuclei. Previous parameterizations are explicit and based on splines [43, 44] or spherical harmonics [68, 69]. Implicitly parameterized DSMs were not used.

The segmentation result of the DSM-based approaches mentioned above heavily depends on the initialization, since none of them yields globally optimal solutions.

In this chapter, we introduce a new globally optimal approach for cell segmentation in microscopy images, which uses implicitly parameterized DSMs based on a linear parameterization. Our approach intrinsically copes with non-elliptical shapes and jointly exploits shape and intensity information by convex energy minimization. Neither prior image binarization nor prior object detection are required. The approach is based on three main contributions:

1. We propose *implicitly* parameterized DSMs for cell segmentation and show that the proposed parameterization leads to a *convex* energy for model fitting. Implicit parameterizations are advantageous, since model fitting can often be performed by convex optimization (cf. Section 2.3.2). Minimization of the convex energy determines the global solution independently of the initialization, is fast, and robust.

2. We introduce a novel iterative global energy minimization method, which *jointly* performs cell segmentation and cluster splitting. The method exploits the inherent *superadditivity* property, simultaneously fits multiple models to the image data, and provably determines a solution close to global optimality. The superadditivity property leverages the lower bound of the energy of the union of models and improves the computational efficiency.

3. We also derive a closed-form solution of the global minimization for non-clustered cell nuclei, which is based on the superadditivity property. This further improves the efficiency since iterative minimization is not required.

The core idea of our approach is to consider the *infimum* of the convex energy for a DSM as a *set energy function*, i.e. a function of the set of image regions where model fitting is performed. We determine optimal regions for fitting and show that for these regions the computation of set energy functions amounts to *convex energy minimization*. To perform joint cell segmentation and cluster splitting, we show that the set energy functions are *superadditive* for disjoint image regions. This structural property is established via the *set-packing polytope* (see Definition 2.3 in Section 2.2.6) and leads to a *necessary optimality condition* for image regions, meaning that the optimal image regions can be determined by only considering the subset of all possible image regions, which pass the condition. We exploit the inherent property of superadditivity to develop a novel and computationally efficient global energy minimization method, which *iteratively* determines the optimal regions. In addition, we derive a *closed-form* solution of the proposed global minimization, which directly determines optimal regions for non-clustered cell nuclei (without requiring iteration). Our energy minimization method does not suffer from local minima and scale-related hyperparameters are automatically determined to facilitate application to image data with different scales. The proposed approach intrinsically copes with intensity inhomogeneities and partial object overlap since shape and intensity information are used jointly. We denote the proposed approach as SuperDSM since it leverages superadditivity and DSMs.

In contrast to previous approaches (see Section 2.3), SuperDSM neither requires an object interaction model (e.g., [40, 49, 62, 63, 64, 67, 77, 78]), nor prior image binarization (e.g., [32, 52, 65, 66, 67]), nor prior object detection (e.g., [41, 42, 43, 44, 56, 57, 58, 59, 61, 77, 117]). In contrast to our previous GOCELL approach (Chapter 5), which is limited to elliptical models, the proposed approach copes with more general shapes by using DSMs, is more efficient since it exploits the

property of superadditivity for energy minimization, and is scale invariant. None of the previous methods mentioned above exploited the superadditivity property. To the best of our knowledge, the proposed approach is the first that combines convex optimization with DSMs for cell segmentation.

We have evaluated our approach using fluorescence microscopy datasets of five different cell types comprising various challenges, including publicly available benchmark datasets, and performed a quantitative comparison with previous methods. It turns out that the proposed approach generally yields competitive or improved results. In addition, we also demonstrate the applicability of our approach to another imaging modality, namely histopathology images with H&E-stained cell nuclei.

This chapter is organized as follows. Section 6.2 introduces the implicitly parameterized DSMs, the corresponding convex energy, and the global energy minimization method which exploits the superadditivity property. Section 6.3 describes the proposed cell segmentation approach including the pre-processing scheme, the automatic choice of hyperparameters for scale invariance, and the post-processing scheme. Section 6.4 provides experimental results and a comparison with previous methods. We discuss the results of our work in Section 6.5.

The work has been submitted for publication [13].

6.2 Superadditivity and convex optimization for segmentation

An overview of the proposed SuperDSM approach for cell nuclei segmentation using deformable shape models is shown in Figure 6.1. The approach consists of four main steps: 1) Pre-processing (scale estimation, determination of intensity offsets, and detection of regions of possibly clustered objects), 2) coarse-to-fine region analysis (computation of the universe of image regions and the corresponding adjacency graph), 3) global energy minimization using deformable shape models, and 4) post-processing. Step 3 is most important and concerns our main contributions.

Below, we describe the proposed global energy minimization method. We first introduce the implicitly parameterized deformable models (Section 6.2.1) and the corresponding convex energy (Section 6.2.2). Then, we describe the superadditive set energy functions (Section 6.2.3), the global optimization objective (Section 6.2.4), and the iterative method for cell segmentation and cluster splitting using global energy minimization, as well as the closed-form solution for non-clustered cell nuclei (Section 6.2.5).

6.2.1 Implicit shape parameterization

We use the zero-level set of a model function to represent the shape of an object. Our model function consists of a polynomial and local deformations. We use a

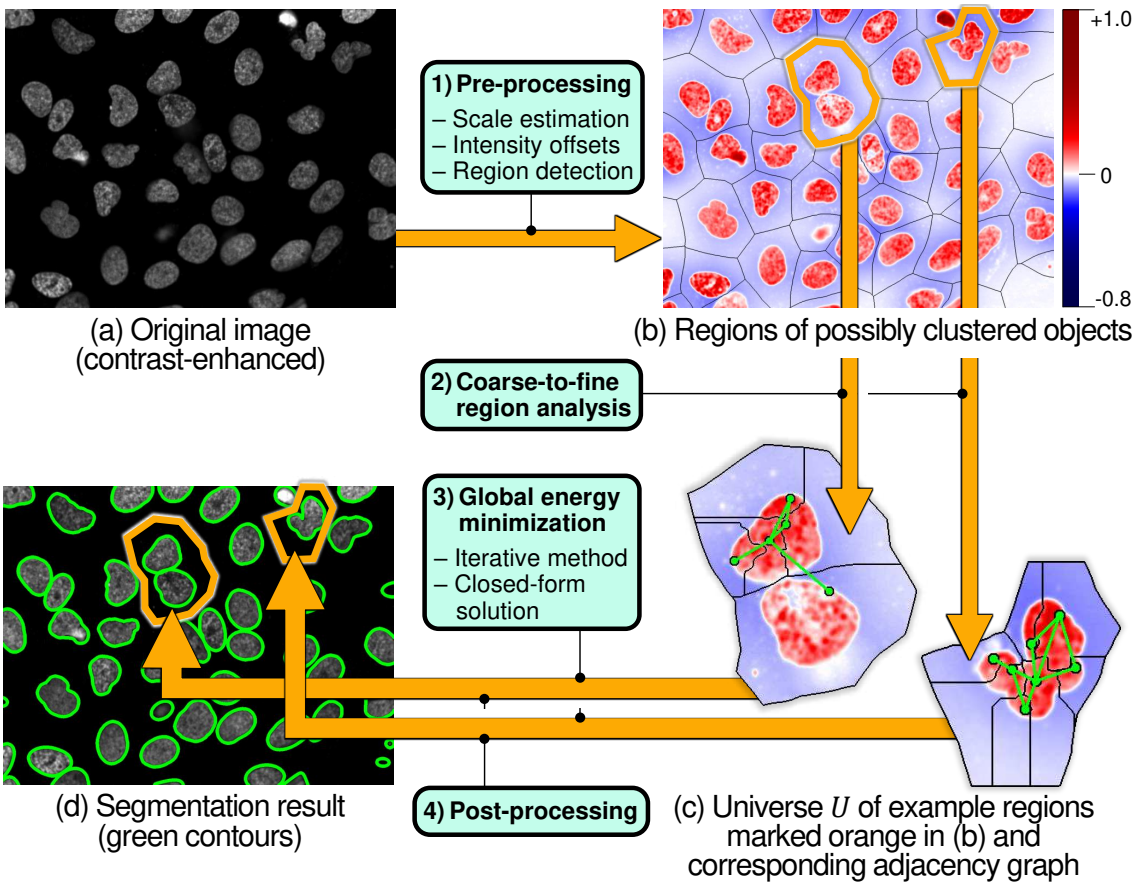


Figure 6.1. Overview of our SuperDSM method for cell nuclei segmentation.

second-order polynomial $s\colon \Omega \to \mathbb{R}$, which maps each image point $x = (x_1, x_2)$ to a real value, where $x \in \Omega$ and $\Omega \subset \mathbb{R}^2$ are all points of an image. We employ the parameterization

$$s\left(x;\theta\right) = \left\langle f_x, \theta \right\rangle, \qquad \theta \in \mathbb{R}^6, \qquad f_x^\top = \begin{bmatrix} x_1^2 & x_2^2 & 2x_1x_2 & x_1 & x_2 & 1 \end{bmatrix}, \tag{6.1}$$

where θ are the polynomial parameters. The zero-level set of Eq. (6.1) corresponds to a conic section, which is limited to elliptical shapes and a few degenerated shapes (e.g., hyperbolic). We consider an image region $\omega \subseteq \Omega$, i.e. a non-empty subset of the image points Ω in an arbitrary but fixed order $\omega = \left\{x^{(1)}, \ldots, x^{(\#\omega)}\right\}$, where # denotes cardinality. Then,

$$S_\omega\left(\theta, \mathbb{0}\right) = F_\omega^\top \cdot \theta, \qquad \text{where } F_\omega = \begin{bmatrix} f_{x^{(1)}} & \cdots & f_{x^{(\#\omega)}} \end{bmatrix}, \tag{6.2}$$

describes a *polynomial surface* within the image region ω, where $\mathbb{0}$ is a vector of zeros with arbitrary dimension (used for notational consistency). The parameterization in Eq. (6.1) was used in the previous GOCELL approach (Chapter 5) to describe *elliptical* shapes.

To represent more general non-elliptical shapes, we augment the polynomial surface by integrating local deformations. We represent the deformations by the smooth perturbation term $G_\omega \cdot \xi$ and define an *implicit deformable* shape model in an image region ω as

$$S_\omega\left(\theta, \xi\right) = F_\omega^\top \cdot \theta + G_\omega \cdot \xi, \qquad \xi \in \mathbb{R}^{\#\Omega}. \tag{6.3}$$

The $\#\omega \times \#\Omega$ matrix G_ω is a block Toeplitz matrix, where each row represents a Gaussian function with standard deviation σ_G centered at the image points $x^{(1)}, \ldots, x^{(\#\omega)}$. The term $G_\omega \cdot \xi$ thus corresponds to a linear combination of Gaussian functions, and ξ are the deformation parameters (weights of the Gaussian functions). The deformable model in Eq. (6.3) includes the elliptical model in Eq. (6.2) as a special case for $\xi = \mathbb{0}$. The implicit parameterization in Eq. (6.3) has the advantage that it is linear in the model parameters θ, ξ, which leads to a *convex* energy. Thus, minimization yields the global solution and can be performed efficiently (Section 6.2.2).

Any pair of model parameters θ, ξ induces two disjoint image regions, that are the zero-sublevel set $\Omega_S^-\left(\theta, \xi\right)$ of the deformable shape model $S_\omega\left(\theta, \xi\right)|_{\omega=\{x\}}$ as a function of $x \in \Omega$ and its corresponding zero-superlevel set $\Omega_S^+\left(\theta, \xi\right)$,

$$\begin{aligned} \Omega_S^-\left(\theta, \xi\right) &= \left\{ x \in \Omega \;\middle|\; S_\omega\left(\theta, \xi\right)|_{\omega=\{x\}} < 0 \right\}, \\ \Omega_S^+\left(\theta, \xi\right) &= \left\{ x \in \Omega \;\middle|\; S_\omega\left(\theta, \xi\right)|_{\omega=\{x\}} > 0 \right\}. \end{aligned} \tag{6.4}$$

These two regions correspond to the interior and exterior of the model, respectively.

6.2.2 Convex energy minimization

We use $g_x \in \mathbb{R}$ to denote the image intensity at an image point $x \in \Omega$. Given the image intensities $g_{x^{(1)}}, \ldots, g_{x^{(\#\omega)}}$, we assume local intensity offsets $\tau_{x^{(1)}}, \ldots, \tau_{x^{(\#\omega)}}$ so that

$$Y_\omega^\top = \begin{bmatrix} g_{x^{(1)}} - \tau_{x^{(1)}} & \ldots & g_{x^{(\#\omega)}} - \tau_{x^{(\#\omega)}} \end{bmatrix} \tag{6.5}$$

defines a coarse subdivision of the image into a set of points Ω_Y^- corresponding to the background and the image points Ω_Y^+ corresponding to the foreground,

$$\begin{aligned} \Omega_Y^- &= \left\{ x \in \Omega \;\middle|\; Y_\omega|_{\omega=\{x\}} < 0 \right\}, \\ \Omega_Y^+ &= \left\{ x \in \Omega \;\middle|\; Y_\omega|_{\omega=\{x\}} > 0 \right\}. \end{aligned} \tag{6.6}$$

The intuition is that $g_x - \tau_x < 0$ (i.e. $x \in \Omega_Y^-$) indicates that an image point x belongs to the background, and $g_x - \tau_x > 0$ (i.e. $x \in \Omega_Y^+$) indicates that an image point x belongs to the foreground. Since cell nuclei in fluorescence microscopy images correspond to bright intensity regions compared to the background, the offsets τ_x can be determined, for example, by Gaussian filtering. In our implementation, we have developed a more sophisticated two-step scheme, which interpolates between Gaussian filtering of the original and clipped intensity values to better cope with boundary points, and is more robust to intensity inhomogeneities (for details see Appendix A).

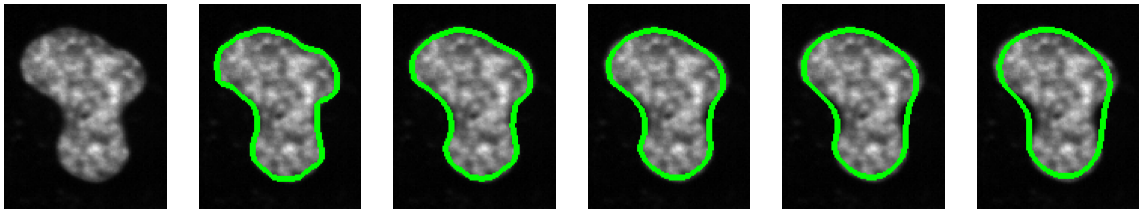
To fit the implicitly parameterized deformable shape model to the image data, we seek to determine the model parameters θ and ξ so that $\omega \cap \Omega_Y^+ \approx \omega \cap \Omega_S^+(\theta, \xi)$ and $\omega \cap \Omega_Y^- \approx \omega \cap \Omega_S^-(\theta, \xi)$. More formally, we minimize the cardinality of the intersections $\omega \cap \Omega_Y^+ \cap \Omega_S^-(\theta, \xi)$ and $\omega \cap \Omega_Y^- \cap \Omega_S^+(\theta, \xi)$,

$$\inf_{\theta, \xi} \psi_\omega(\theta, \xi), \qquad \psi_\omega(\theta, \xi) = \ell_\omega(\theta, \xi) + \alpha \cdot \|\xi\|_1, \tag{6.7}$$

using an L_1 regularization for the deformation parameters ξ, where $\ell_\omega(\theta, \xi)$ corresponds to the cardinality of the intersections. Direct minimization of the cardinality amounts to the minimization of the $0/1$ loss. However, this is challenging since the $0/1$ loss is neither smooth nor convex. Analogously to Chapter 5, we thus use the surrogate loss function ϕ_1 from Eq. (4.4) for the cardinality of the intersections, i.e.

$$\begin{aligned} \ell_\omega(\theta, \xi) &= \sum_{x \in \omega} \phi_1\left(Y_\omega|_{\omega=\{x\}}; S_\omega(\theta, \xi)|_{\omega=\{x\}} \right) \\ &= \left\langle \mathbb{1}_{\#\omega}, \ln\left(1 + \exp\left(-Y_\omega \odot S_\omega(\theta, \xi)\right)\right) \right\rangle, \end{aligned} \tag{6.8}$$

where "ln" and "exp" are defined component-wise, "$\odot$" is the Hadamard product, and $\mathbb{1}_{\#\omega}$ is a vector of ones with dimension $\#\omega$. Using the surrogate loss is



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

Figure 6.2. Example segmentation results for different values of the regularization parameter α. (a) Original image section. (b) Segmentation result using $\alpha = 0.001$, (c) $\alpha = 0.002$, (d) $\alpha = 0.003$, (e) $\alpha = 0.004$, (f) $\alpha = 0.005$.

advantageous since Eq. (6.8) is *convex*. Another advantage of Eq. (6.8) is that the image intensities are directly exploited via Y_ω and image binarization is not required. The parameter $\alpha \geq 0$ in Eq. (6.7) governs the regularization of the deformations. Example segmentation results for different values of α for an image section of U2OS cells are shown in Figure 6.2. The section shows a single cell nucleus (according to the ground truth from [103]) and has a size of 136×108 pixels. Increasing α leads to a smoother segmentation result.

Eqs. (6.7) and (6.8) correspond to an unconstrained convex problem (see Appendix B.1 for a proof). To solve this problem, we use a fast second-order method based on consecutive Newton steps (see Section 2.2.5). Due to convexity, the method determines the globally optimal solution for θ and ξ independently of the initialization. We use the implementation [121] of the second-order method and a specific initialization scheme for faster convergence (described in Appendix C.3).

6.2.3 Set energy functions and superadditivity

The implicit deformable shape model introduced above represents a *single* object. For globally optimal model fitting for an entire image, we exploit that linearly parameterized single-object models such as Eq. (6.2) and Eq. (6.3) naturally generalize to the *multi-object* case (Chapter 5). Let the set U be a *universe* of region fragments, where each region fragment is a connected image region comprising image points of at most a single object (and the image background). The objective then is to determine a *low-cardinality* and *minimal-energy family* $\mathscr{X}$ of sets of the region fragments U, subject to the constraint that $\bigcup \mathscr{X} = U$. The energy of a set $X \subseteq U$ of image regions is given by the solution of Eq. (6.7) for $\omega = \tilde{\omega}(X) \cup \Omega_Y^-$,

$$\inf_{\theta,\xi} \psi_{\tilde{\omega}(X)\cup\Omega_Y^-}(\theta,\xi), \qquad \text{where } \tilde{\omega}(X) = \bigcup X \tag{6.9}$$

is the model activity region defined by the set X of region fragments[5] (cf. Section 5.3.1), and $\bigcup X$ are all image points of the set X. In Chapter 5, this result was used for elliptical models, but determining the optimal family $\mathscr{X}$ required the computation of *all* admissible sets using prior assumptions (e.g., maximum cardinality of U) to maintain computational tractability. In this work, we exploit the result for deformable shape models and for the property of *superadditivity*. Superadditivity denotes the property that the energy of any set X is lower-bounded by the sum of energies of its disjoint subsets. This has the advantage that the optimal family $\mathscr{X}$ can be determined by only considering a *subset* of all possible image regions. The proposed global optimization method for deformable shape models is far more sophisticated than the one for elliptical models, since we automatically confine the computations to a meaningful subset of the admissible sets, using the analytical property of superadditivity instead of requiring prior assumptions.

Below, we first formally define the universe U of region fragments, formulate Eq. (6.9) as a *set energy function*, and derive its property of superadditivity. We then exploit this property to formally define a suitable optimization objective for the optimal solution $\mathscr{X}$ (Section 6.2.4). Minimization of the obtained objective is **NP**-hard and thus computationally challenging. However, we further leverage superadditivity and convexity to decompose the challenging optimization problem into easily solvable sub-problems (Section 6.2.5).

Let $\mathcal{E} \subseteq U \times U$ represent adjacent region fragments, i.e. $\{u, v\} \in \mathcal{E}$ if and only if u is adjacent to v, using the following definition of adjacency:

Definition 6.1 (Adjacency). Two region fragments $u, v \in U$ are considered *adjacent* if and only if $\Omega_Y^+ \cap (u \cup v)$ contains a path between u and v. $\Pi \subseteq U \times U$ represents the *connected* region fragments (i.e. $\{u, v\} \in \Pi$ if and only if the adjacency graph $\mathcal{G} = (U, \mathcal{E})$ contains a path between the regions u and v).

Figure 6.1c shows an example universe of region fragments (black lines) and the corresponding adjacency graph (green lines) for two connected components.

In order to formally introduce set energy functions (see below), we first define the family of all connected subsets of U with cardinality k or less,

$$\mathbb{P}_k(U) = \{X \subseteq U | \#X \leq k, X \times X \subseteq \Pi\}. \tag{6.10}$$

In the following, we use $\mathbb{P}(U) = \mathbb{P}_{\#U}(U)$ as a short form.

The objective function in Eq. (6.9) is defined by the energy ψ_ω in Eq. (6.7) for the image region $\omega = \tilde{\omega}(X) \cup \Omega_Y^-$. Image points within large regions of image background generally yield low energy values and are thus negligible. Since the

[5]The symbol X is slightly differently defined in this chapter than in Chapter 5. In Chapter 5, the *union* of region fragments was denoted by X. In this chapter, X denotes a *set* of region fragments and $\tilde{\omega}(X)$ denotes their union.

set Ω_Y^- mostly contains such image points, we have

$$\inf_{\theta,\xi} \psi_{\tilde{\omega}(X)\cup\Omega_Y^-}(\theta,\xi) \approx \inf_{\theta,\xi} \psi_{\tilde{\omega}(X)}(\theta,\xi). \tag{6.11}$$

This gives rise to the *set energy function* $c\colon \mathbb{P}(U) \to \mathbb{R}_+$,

$$c(X) = \inf_{\theta,\xi} \psi_{\tilde{\omega}(X)}(\theta,\xi). \tag{6.12}$$

Below, we describe a relation of Eq. (6.12) to the set-packing polytope, which we use to establish the property of superadditivity.

For any family of sets $X_1, \ldots, X_m \subseteq U$, the set $P(X_1, \ldots, X_m)$ of solutions $\eta \in \mathbb{R}_+^m$ for the inequality

$$\sum_{k\in[m]} [u \in X_k] \cdot \eta_k \leq 1 \quad \text{for all } u \in U, \tag{6.13}$$

is a *set-packing polytope* (the polytope associated with the *max-weight set-packing* problem and its linear relaxation, cf. Definition 2.3), using the Iverson brackets defined by [statement] = {1 if statement is true; 0 else}. Then, *any* family of sets $X_1, \ldots, X_m$ and associated weights $\eta \in P(X_1, \ldots, X_m)$ yields a lower bound of the set energy function in Eq. (6.12) for the set $X_1 \cup \cdots \cup X_m$ (Property B.2 in Appendix B.2).

When the sets $X_1, \ldots, X_m$ are *disjoint*, a vector of ones with dimension m is always contained in $P(X_1, \ldots, X_m)$ and thus the set energy function c is *superadditive*,

$$c(X_1) + \cdots + c(X_m) \leq c(X_1 \cup \cdots \cup X_m). \tag{6.14}$$

Thus, for any two disjoint, non-empty sets $A, B \subset U$, the sum of their energies $c(A) + c(B)$ is a *lower bound* of the energy of their union, $c(A \cup B)$. This means that the energy c of a set can be directly deduced (without optimization). Moreover, the *singleton* set $\{u\}$ (i.e. set with exactly one element) of any element $u \in U$ is the set with the *lowest* energy among all those containing the element u. In terms of energy minimization, this means that *any* image is best fitted by the singleton sets of U. This likely leads to over-segmentation and is thus not well-suited. Below, we describe an extension of the set energy functions which avoids over-segmentation.

6.2.4 Extended set energy functions and optimization objective

To obtain a meaningful segmentation result and avoid over-segmentation, we extend the set energy functions in Eq. (6.12). We add the constant term $\beta \geq 0$ and define the *extended set energy function* $\tilde{c}\colon \mathbb{P}(U) \to \mathbb{R}_+$,

$$\tilde{c}(X) = c(X) + \beta. \tag{6.15}$$

In contrast to the original set energies c, the extended energies $\tilde{c}(A) + \tilde{c}(B)$ of two disjoint, non-empty sets $A, B \subset U$ can actually be *higher* than the extended energy of their union, $\tilde{c}(A \cup B)$, since $\tilde{c}(A) + \tilde{c}(B) - \tilde{c}(A \cup B) \leq \beta$ due to Eq. (6.14). Thus, β is the maximum allowed energy difference of merging A and B. Only if the energy $c(A \cup B)$ exceeds $c(A) + c(B)$ by less than β, merging A and B is beneficial in terms of energy minimization using the extended set energy $\tilde{c}$. Merging A and B corresponds to using a single deformable shape model for the union $A \cup B$ instead of two separate shape models for A and B.

Using the linear program (LP) relaxation of the *max-weight set-packing* problem (cf. Section 2.2.6)

$$\textsc{MaxSetPacking}_{\text{LP}}(\mathscr{S}) = \max_{\eta \in P(\mathscr{S})} \sum_{k \in [m]} \eta_k \cdot c(X_k), \quad \text{where } \mathscr{S} = X_1, \ldots, X_m \tag{6.16}$$

and the set-packing polytope P is the set of solutions of the inequality in Eq. (6.13), we obtain the following lower bound of the extended set energy $\tilde{c}(X)$:

Property 6.1. *Given a set $X \subset U$ with cardinality $\#X = k + 1 \geq 2$, the rhs of*

$$\tilde{c}(X) \geq \textsc{MaxSetPacking}_{\text{LP}}(\mathbb{P}_k(X)) + \beta \tag{6.17}$$

is a lower bound of its extended set energy (lhs).

Proof. *See Appendix B.3.* □

To define the optimization objective for global energy minimization, we consider the overall minimal energy for a subset of regions $\mathscr{X} \subseteq \mathbb{P}(U)$, which covers the whole universe U. Using the extended set energy from Eq. (6.15), this formally corresponds to $\textsc{MinSetCover}(\mathbb{P}(U))$ where

$$\textsc{MinSetCover}(\mathscr{S}) = \min_{\mathscr{X} \subseteq \mathscr{S}} \sum_{X \in \mathscr{X}} \tilde{c}(X) \quad \text{subject to } \bigcup \mathscr{S} = \bigcup \mathscr{X}, \tag{6.18}$$

which is an instance of the *min-weight set-cover* problem (cf. Definition 2.2). Computation of $\textsc{MinSetCover}(\mathbb{P}(U))$ is challenging for two reasons. First, Eq. (6.18) is **NP**-hard. To cope with this, we use an approximation algorithm which determines the global solution within a tight approximation ratio (see Section 6.3.3). Second, $\mathbb{P}(U)$ has a potentially large cardinality. We address this by avoiding the computation of the *whole* family $\mathbb{P}(U)$, as described below.

6.2.5 Global optimization scheme

To cope with the potentially large cardinality of $\mathbb{P}(U)$, we are interested in a criterion for a set $X \subseteq U$ which guarantees that the set X is *negligible*, i.e. $\textsc{MinSetCover}(\mathbb{P}(U)) = \textsc{MinSetCover}(\mathbb{P}(U) \setminus \{X\})$. Excluding such sets ultimately yields a subset $\mathscr{U} \subseteq$

$\mathbb{P}(U)$, which satisfies $\textsc{MinSetCover}(\mathscr{U}) = \textsc{MinSetCover}(\mathbb{P}(U))$ but is of lower cardinality than $\mathbb{P}(U)$. We derive such a criterion from the following *lower bound* of the global optimization objective:

Property 6.2. *Let $X \subseteq U$ be a set with cardinality $\#X = k+1 \geq 2$. If X or a superset of X are* not negligible*, i.e.*

$$\exists Y : X \subseteq Y \subseteq U \wedge \textsc{MinSetCover}(\mathbb{P}(U)) < \textsc{MinSetCover}(\mathbb{P}(U) \setminus \{Y\}), \quad (6.19a)$$

then the lhs of

$$\tilde{c}(X) + \sum_{u \in U \setminus X} c(\{u\}) \leq \textsc{MinSetCover}(\mathbb{P}(U)) \quad (6.19b)$$

is a lower bound of the global optimization objective (rhs).

Proof. *See Appendix B.3.* □

In addition to the lower bound established above in Eq. (6.19b), the monotonicity $\mathbb{P}_k(U) \subseteq \mathbb{P}_{k+1}(U)$ yields *upper bounds* of $\textsc{MinSetCover}(\mathbb{P}(U))$,

$$\begin{aligned} &\textsc{MinSetCover}(\mathbb{P}(U)) = \\ &\textsc{MinSetCover}(\mathbb{P}_{\#U}(U)) \quad \leq \ldots \leq \quad \textsc{MinSetCover}(\mathbb{P}_1(U)) \\ &\qquad\qquad\qquad\qquad\qquad\qquad = \sum_{u \in U} \tilde{c}(\{u\}). \end{aligned} \quad (6.20)$$

Note that a set $X \subseteq U$ is negligible if and only if $X \notin \mathscr{X}$, where $\mathscr{X}$ is the family of optimal sets in Eq. (6.18) which solve $\textsc{MinSetCover}(\mathbb{P}(U))$. Eq. (6.20) relaxes the rhs of Eq. (6.19b), which is *necessary* for $X \in \mathscr{X}$ (but not sufficient). Thus, Property 6.2 in conjunction with Eq. (6.20) can be seen as a *necessary optimality condition* for a set X. By negating this condition (i.e. considering the logical complement), we obtain the following criterion to identify negligible sets:

Criterion 6.1 (Negligible sets). *Let $X \subseteq U$ a set with cardinality $\#X = k+1 \geq 2$. If*

$$\tilde{c}(X) > \textsc{MinSetCover}(\mathbb{P}_k(U)) - \sum_{u \in U \setminus X} c(\{u\}), \quad (6.21a)$$

then X and its supersets are negligible*, i.e.*

$$\textsc{MinSetCover}(\mathbb{P}(U)) = \textsc{MinSetCover}(\mathbb{P}(U) \setminus \{Y\}) \quad (6.21b)$$

for all $Y : X \subseteq Y \subseteq U$.

Proof. *See Appendix B.3.* □

Algorithm 6.1: Iterative solution of MinSetCover ($\mathbb{P}(U)$).

input: Adjacency graph $\mathcal{G} = (U, \mathcal{E})$

1 `iter1` $\leftarrow \{\{u\} | u \in U\}$; *// initialize* $\mathscr{U}_1 = \mathbb{P}_1(U)$
2 $\mathscr{U} \leftarrow$ `iter1`; *// sets for which* $\tilde{c}(X)$ *was computed*
3 **do** *// iterate* $\mathscr{U} = \mathscr{U}_1, \mathscr{U}_2, \ldots$
4 `value` $\leftarrow$ MinSetCover ($\mathscr{U}$); *// optimization objective*
5 `iter0` $\leftarrow$ family of all sets $X \cup \{u\}$ where $X \in$ `iter1`, $u \in U \setminus X$, and $\exists v \in X, \{u, v\} \in \mathcal{E}$;
6 `iter1` $\leftarrow \{\}$;
7 **for** $X \in$ `iter0` **do**
8 $\tilde{c}_{\max} \leftarrow$ `value` $- \sum_{u \in U \setminus X} c(\{u\})$; *// Property 6.2*
9 $\tilde{c}_{\min} \leftarrow \beta +$ MaxSetPacking$_{\mathrm{LP}}$ $\{Y \in \mathscr{U} | Y \subset X\}$; *// Property 6.1*
10 **if** $\tilde{c}_{\min} \leq \tilde{c}_{\max}$ **then** *// Criterion 6.1*
11 compute $\tilde{c}(X)$ and insert X into $\mathscr{U}$;
12 **if** $\tilde{c}(X) \leq \tilde{c}_{\max}$ **then** *// Criterion 6.1*
13 `iter1` $\leftarrow$ `iter1` $\cup \{X\}$;
14 **until** #`iter1` $= 0$;
15 **return** family $\mathscr{X}$ corresponding to "`value`", cf. Eq. (6.18)

To compute MinSetCover ($\mathbb{P}(U)$), we consider the sequence $\mathscr{U}_1, \ldots, \mathscr{U}_{\#U}$, where each subset $\mathscr{U}_k \subseteq \mathbb{P}_k(U)$ is obtained by excluding sets according to Criterion 6.1. This procedure guarantees MinSetCover($\mathscr{U}_{\#U}$) = MinSetCover($\mathbb{P}(U)$) due to Eq. (6.21b) and is formally described in Algorithm 6.1. First, Property 6.1 and Property 6.2 are used to determine a lower bound $\tilde{c}_{\min}$ (line 9) and an upper bound $\tilde{c}_{\max}$ (line 8) of the extended set energy $\tilde{c}(X)$. The set is excluded if the lower bound exceeds the upper bound (line 10). Otherwise, $\tilde{c}(X)$ is computed (line 11) and the set X is excluded if $\tilde{c}(X)$ exceeds the upper bound $\tilde{c}_{\max}$ (line 12). The lower and upper bounds $\tilde{c}_{\min}$ and $\tilde{c}_{\max}$ are tightened from iteration to iteration (due to monotonic increase of the family $\mathscr{U}$ and monotonic decrease of the variable `value`). Thus, if a set X is excluded, all supersets $Y \supset X$ are also excluded in subsequent iterations and computation of $\tilde{c}(Y)$ is not required due to Property 6.1. The number of iterations is upper-bounded by the cardinality of the universe. An example run-through of Algorithm 6.1 is given in Appendix C.1.

Compared to classically tree-based branch-and-bound schemes, Algorithm 6.1 builds *multiple* trees $\mathcal{H}$ along the edges of the adjacency graph $\mathcal{G} = (U, \mathcal{E})$, each rooted in the singleton sets of U. This corresponds to the directed acyclic graph (DAG)

$$\mathcal{H} = (\mathscr{U}, \mathcal{E}'), \qquad \text{where } (X, Y) \in \mathcal{E}' \text{ iff } X \subset Y. \tag{6.22}$$

The graph $\mathcal{H}$ comprises only the subset $\mathscr{U}$ of the admissible nodes $\mathbb{P}(U)$. Also, in contrast to previous DAG-based approaches (e.g., [93, 122]), the segmentation is *not*

encoded in our graph structure. These two properties naturally lead to comparably *shallow* graphs, thus, neither heuristic pruning [122] nor prior assumptions [12] are required to obtain graphs of computationally tractable size.

Algorithm 6.1 can be interpreted that it excludes sets corresponding to falsely merged objects (see the example in Appendix C.1). However, if the *whole* universe U corresponds to a *single* object, i.e. if $\tilde{c}(U) = \textsc{MinSetCover}(\mathbb{P}(U))$, then no falsely merged objects can possibly occur. In this case, Eq. (6.19b) is fulfilled for any $X \subseteq U$ due to Eq. (6.14), and the computational cost (cardinality of $\mathscr{U}$) grows to $\#\mathbb{P}(U)$. To avoid this, we introduce Criterion 6.2, which identifies this case a priori and provides a closed-form solution:

Criterion 6.2 (Closed-form solution of $\textsc{MinSetCover}(\mathbb{P}(U))$). *If* $\tilde{c}(U) \leq 2\beta + \sum_{u \in U} c(\{u\})$, *then* $\textsc{MinSetCover}(\mathbb{P}(U)) = \tilde{c}(U)$.

Proof. *See Appendix B.3.* □

Criterion 6.2 is applied for direct segmentation of non-clustered cell nuclei without using the iterative Algorithm 6.1 (see Section 6.3). Note that Criterion 6.2 can also be seen as a *sufficient optimality condition* for the set U (as opposed to the *necessary* optimality condition which is the basis for Criterion 6.1).

The iterative Algorithm 6.1 is computationally more efficient for *small* values of β (Criterion 6.1 then excludes more sets). On the other hand, the closed-form solution (Criterion 6.2) is more efficient for *large* values of β (since in this case the margin of the inequality in Criterion 6.2 is larger).

6.3 Cell nuclei segmentation using superadditivity, convex optimization, and deformable shape models

The proposed SuperDSM method for cell nuclei segmentation consists of four main steps (cf. Figure 6.1). First, an image is pre-processed to estimate the scale of the objects, determine the image intensity offsets, and detect image regions corresponding to possibly clustered objects. Second, a coarse-to-fine region analysis is performed to compute the universe U of region fragments and the corresponding adjacency graph. Third, global energy minimization is performed by iterative (Algorithm 6.1) and direct (Criterion 6.2) solution of $\textsc{MinSetCover}(\mathbb{P}(U))$ in Eq. (6.18) above. This includes the automatic choice of hyperparameters for scale invariance, comprising the weight α of the regularization of the deformations in Eq. (6.7) and the constant term β of the extended set energy functions in Eq. (6.15). Both parameters are determined based on the scale σ, which is computed automatically in the first step. Fourth, post-processing is performed. The steps are detailed below.

6.3.1 Scale estimation, intensity offsets, and detection of regions of possibly clustered objects

To estimate the scale of cell nuclei, we compute the Hessian matrix of the image intensities [123] and determine local maxima of the determinant of the Hessian in scale space [108]. False-positive detections are discarded if *(i)* the detection corresponds to a non-negative response of the Laplacian of Gaussian filter or *(ii)* the scale of the detection is an outlier (determined based on the mean absolute difference to the median scale). The mean of the remaining inliers is then associated with the scale σ of an image.

The intensity offsets τ_x for all image points $x \in \Omega$ are computed using modified Gaussian filtering with standard deviation according to the estimated scale σ (see Supplemental Material 1 in [13]). The denoised image intensities g_x (obtained by a Gaussian filter with standard deviation $\sqrt{2}$) and the intensity offsets τ_x are then used to compute γ_Ω (image intensities with τ_x offset) for all image points $x \in \Omega$ according to Eq. (6.5) by setting $\omega = \Omega$.

We also determine the universe U of region fragments and the corresponding adjacency graph $\mathcal{G} = (U, \mathcal{E})$. In general, $\mathcal{G}$ is disconnected due to Definition 6.1 (Section 6.2.3). Each *connected* component corresponds to a *region of possibly clustered objects*. For computational efficiency, we first determine these regions, and then the corresponding *connected* adjacency graph $\mathcal{G}$ for each region of possibly clustered objects (formally this is the same as considering the disconnected adjacency graph of the whole image). We consider the connected components of the foreground region Ω^+_γ, defined as in Eq. (6.6). This generally yields several components which correspond to the background due to image noise. We identify such components by considering the perimeter-to-area ratio (P/A ratio). The intuition is that components due to image noise have strongly irregular contours (jagged or wavelike), while isolated and clustered cell nuclei have rather smooth contours. Components with strongly irregular contours can be identified by a large perimeter compared to the area. We discarded such image regions when the P/A ratio is larger than a threshold of 0.2, which was chosen empirically. We then obtain the regions of possibly clustered objects as the regions of the Voronoi diagram of the remaining connected components (cf. Figure 6.1b).

6.3.2 Coarse-to-fine region analysis

We next determine the universe U of region fragments and the corresponding *connected* adjacency graph $\mathcal{G}$ separately for each region of possibly clustered objects (cf. Figure 6.1c). The main requirement for the universe U is that each region fragment overlaps with at most one object, while not generating a universe U of an unnecessarily large cardinality (which would increase the run time of Algorithm 6.1). We thus start with a region of possibly clustered objects as a whole,

determine the *irregularity* of an object in that region, and split the region into smaller parts as long as the irregularity is large, ultimately obtaining the region fragments (which are not split further). To determine the irregularity of an image region ω, we minimize the energy function in Eq. (6.7) for $\xi = \mathbb{0}$ and consider the *normalized energy*

$$r(\omega) = \inf_{\theta} \psi_{\omega}(\theta, \mathbb{0}) / \#\omega. \tag{6.23}$$

Eq. (6.23) corresponds to fitting an elliptical model to the region ω. It is beneficial to use elliptical models here, since the energy of these models is more sensitive to shape irregularities of objects than the energy of deformable shape models. Since the analysis is performed by splitting large image regions into smaller parts, a *coarse-to-fine region analysis* scheme is induced (in contrast to Algorithm 6.1 which uses a fine-to-coarse scheme). See Appendix C.2 for details.

6.3.3 Scale invariant global energy minimization

Each graph $\mathcal{G} = (U, \mathcal{E})$ determined as described in Section 6.3.2 is processed as follows. First, the extended set energies $\tilde{c}(U)$ and $\tilde{c}(\{u\})$ are computed for all region fragments $u \in U$ by solving Eq. (6.7). Second, it is checked whether U corresponds to a non-clustered cell nucleus using the inequality in Criterion 6.2. If it does, the closed-form solution $\mathscr{X} = \{U\}$ is applied. Otherwise, the iterative Algorithm 6.1 is used to determine the solution $\mathscr{X}$ of MinSetCover($\mathbb{P}(U)$) using $\mathcal{G}$. Below, we describe the automatic choice of the hyperparameters α and β for Criterion 6.2 and Algorithm 6.1 to establish scale invariance, and introduce efficient implementations of Algorithm 6.1 and Eq. (6.7).

The extended set energy function $\tilde{c}(X) = c(X) + \beta$ in Eq. (6.15) depends on the hyperparameter $\beta \geq 0$. To properly choose a value for β, we need to understand how $c(X) = \inf_{\theta,\xi} \psi_{\tilde{\omega}(X)}(\theta, \xi)$ in Eq. (6.12) depends on the *scale* σ of an image. For an arbitrary image region $\omega = \tilde{\omega}(X)$, recall that the energy $\inf_{\theta,\xi} \psi_{\omega}(\theta, \xi)$ approximates the cardinality of a set of image points (Section 6.2.2). Since the number of image points corresponds to the area, the number changes quadratically with respect to the scale (for 2-D images). Thus, it is reasonable to assume that the energy $\inf_{\theta,\xi} \psi_{\omega}(\theta, \xi)$ depends quadratically on the scale. Figure 6.3b shows the energy for image sections of different cell types in different datasets generated by Gaussian filtering and sub-sampling at different scales σ. It can be seen that the energy depends quadratically on the scale (note that the vertical axis is scaled quadratically). Thus, it is reasonable to choose $\beta = \beta_{\texttt{factor}} \cdot \sigma^2$ as a quadratic function of the scale to achieve scale invariance. In our experiments using image data of different scales and cell types, $\beta_{\texttt{factor}} = 0.33$ turned out to be a reasonable choice (cf. Section 6.4).

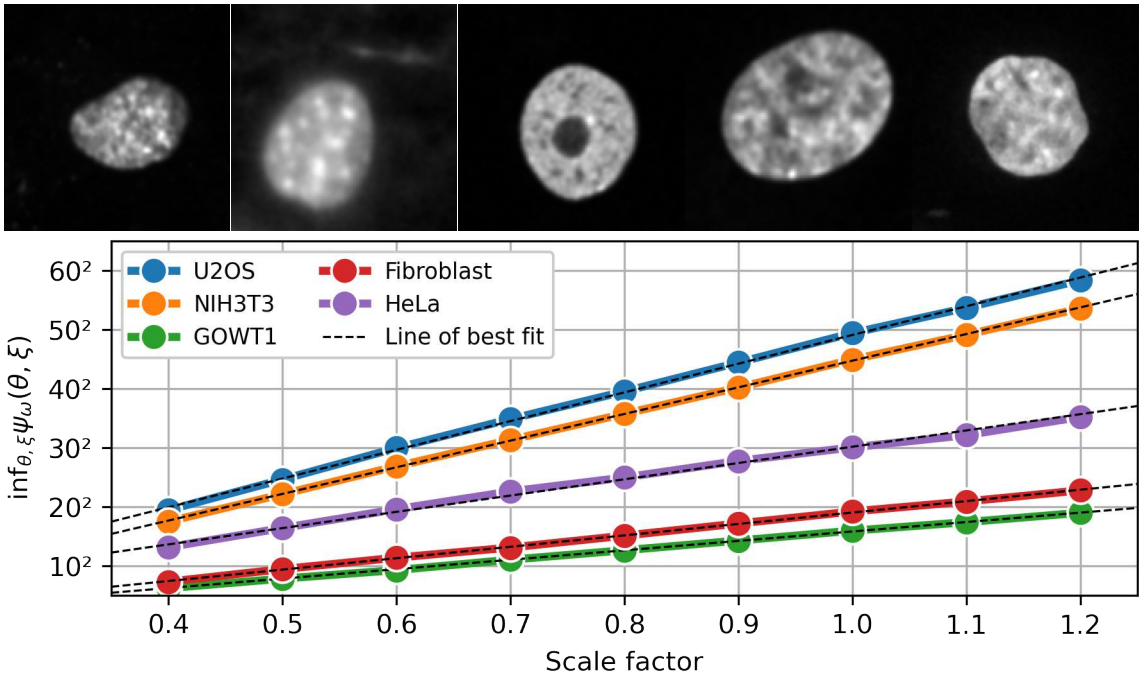


Figure 6.3. Relation of the energy $\inf_{\theta,\xi} \psi_\omega(\theta, \xi)$ and the scale σ. Top: Example image regions of different cell types, from left to right: U2OS, NIH3T3, GOWT1, Fibroblast, HeLa. Bottom: Corresponding energy $\inf_{\theta,\xi} \psi_\omega(\theta, \xi)$ as a function of the scale.

For Algorithm 6.1, the extended set energies $\tilde{c}(X) = \inf_{\theta,\xi} \psi_{\tilde{\omega}(X)}(\theta, \xi) + \beta$ have to be computed often. Each computation amounts to solving the convex problem in Eq. (6.7). Efficient implementation of convex energy minimization is thus crucial. Eq. (6.7) includes the term $G_\omega \cdot \xi$ via Eq. (6.8) and Eq. (6.3), which can be interpreted as radial basis function interpolation [124] due to the block Toeplitz structure of the matrix G_ω. Reducing the number of points (summands of the interpolant) motivates the approximation $G_\omega \cdot \xi \approx \tilde{G}_\omega \cdot \tilde{\xi}$, where the matrix $\tilde{G}_\omega$ is constructed from a subset of the columns of G_ω (the rows are normalized to the sum of 1) and $\tilde{\xi}$ is a vector of lower dimension than ξ (the dimension of ξ corresponds to the overall number of points in an image). The columns of G_ω correspond to the regular grid of all image points Ω, and the columns of $\tilde{G}_\omega$ correspond to the sub-sampled regular grid of image points within the region ω spaced by $2\sigma_G$ (where σ_G is the standard deviation of the Gaussian function used for the matrix G_ω, cf. Section 6.2.1). We used $\sigma_G = 0.2\,\sigma$ (where σ is the scale of an image). Due to sub-sampling, the number of points of the regular grid and thus also the dimension of $\tilde{\xi}$ scales inverse-quadratically with σ. Since the regularization parameter α of the energy in Eq. (6.7) is a factor of $\|\tilde{\xi}\|_1$ (using $\tilde{\xi}$ instead of ξ), inverse-quadratic scaling of the dimension of $\tilde{\xi}$ needs to be compensated by quadratically scaling α. We used $\alpha = \alpha_{\texttt{factor}} \cdot \sigma^2$ and $\alpha_{\texttt{factor}} = 5 \cdot 10^{-4}$ in our experiments (Section 6.4). The approximation $G_\omega \cdot \xi \approx \tilde{G}_\omega \cdot \tilde{\xi}$ concerns only the deformations of the shape representation in Eq. (6.3). The approximated shape representation has a

somewhat lower expressive power (since the matrix $\tilde{G}_\omega$ is of lower rank than G_ω), but substantially increases the computational efficiency, since less parameters need to be determined (the dimension of $\tilde{\xi}$ is lower than ξ). Another interpretation of the perturbation term $G_\omega \cdot \xi$ is that it corresponds to a low-pass filter of ξ, which is due to the block Toeplitz structure of the matrix G_ω and since each row represents a Gaussian function (see above). High frequencies in ξ are suppressed and thus sub-sampling ξ introduces only minor errors. Therefore, minimization using the approximation yields a solution close to the globally optimal solution for the whole segmentation task with the original minimization. Further implementation details of Algorithm 6.1 and an initialization scheme for convex programming with fast convergence are described in Appendix C.3.

Robust approximation of MinSetCover

To solve the **NP**-hard min-weight set-cover problem MinSetCover in Algorithm 6.1, we use the approximative Algorithm 6.2, which iteratively performs a two-step scheme. First, a *greedy* step [27] determines the family $\mathscr{X} \subseteq \mathscr{S}$ for Eq. (6.18) so that the value $\tilde{\psi} = \sum_{X \in \mathscr{X}} \tilde{c}\,(X)$ of the global optimization objective MinSetCover($\mathbb{P}\,(U)$) is at most factor $H_{\#U} = \sum_{k=1}^{\#U} 1/k$ higher than the globally optimal solution (due to false merges/splits, cf. Section 2.2.6). Any false splits that may arise are coped with by the subsequent *merge* step, which merges subsets of $\mathscr{X}$ if this decreases the value $\tilde{\psi}$ of the global optimization objective (cf. Chapter 5). To cope with possible occurrences of false merges, both steps are repeated using a more conservative merging strategy, i.e. decreasing β by a decay factor γ, and the family $\mathscr{X}$ which yields the overall *lowest* value $\tilde{\psi}$ is considered as final solution. We used 5 iterations and $\gamma = 0.8$. The approximation ratio of Algorithm 6.2 is $H_{\#U}$ or better, since modifications of the greedy solution are only permitted if the value $\tilde{\psi}$ is lowered. This conservative upper bound of the approximation ratio can be tightened *a posteriori*. Let $\hat{\psi}$ be the unknown exact solution of Eq. (6.18) and $\hat{\psi}_{\mathrm{LP}} \leq \hat{\psi}$ the exact solution of the LP relaxation. The approximation ratio $\hat{\psi}/\tilde{\psi}$ is thus at worst $\hat{\psi}_{\mathrm{LP}}/\tilde{\psi}$. In our experiments, the average approximation ratio was at least 99.7 % (median: 100.0 %). Moreover, we found that a de-facto exact solution ($\hat{\psi}_{\mathrm{LP}}/\tilde{\psi} \geq 0.99$) was determined in at least 92.1 % of the cases. Figure 6.4 shows lower bounds of the ratio of de-facto exact solutions for different values of γ and iteration numbers. Note that the special case $\gamma = 1$ corresponds to our previous GOCELL approach (Chapter 5). It can be seen that for $\gamma < 1$, higher lower bounds are obtained, thus, our new method improves the global optimality (we used $\gamma = 0.8$ and 5 iterations as indicated in the figure).

Fast approximation of MaxSetPacking$_{\mathrm{LP}}$

Algorithm 6.1 also requires solving MaxSetPacking$_{\mathrm{LP}}$ defined in Eq. (6.16), which can be solved directly (Section 2.2.5). However, the lower bound computed by

Algorithm 6.2: Approximative solution of MinSetCover ($\mathscr{S}$).

input: Family $\mathscr{S} \subseteq \mathbb{P}(U)$; number of iterations `max_iter`; decay factor $\gamma \in (0,1)$

Let $\tilde{c}(X, \beta') = c(X) + \beta'$;
for `iter` $= 1, \ldots,$ `max_iter` **do**
  $\mathscr{X}' \leftarrow \{\}$; $V \leftarrow U$; $\mathscr{Z} \leftarrow \mathscr{S} = \{X_1, \ldots, X_n\}$;
  $\beta' \leftarrow \beta \cdot \gamma^{(\texttt{iter}-1)}$;
  while $\#V > 0$ **do** *// greedy step*
    Set $T_k = \tilde{c}(X_k, \beta') / (X_k \cap V)$ for all $k = 1, \ldots, n$;
    Set $\hat{k} \in [n]$ so that $T_{\hat{k}} = \min_{k \in [n]} T_k$;
    Insert $X_{\hat{k}}$ into $\mathscr{X}'$; and $V \leftarrow V \setminus X_{\hat{k}}$;
  while $\#\mathscr{Z} > 0$ **do** *// merge step*
    Set $\hat{k} \in [n]$ so that $X_{\hat{k}} \in \mathscr{Z}$ and
     $\tilde{c}(X_{\hat{k}}, \beta') = \min\left\{\tilde{c}(X_{\hat{k}}, \beta') \middle| k \in [n], X_k \in \mathscr{Z}\right\}$;
    if $X_{\hat{k}} \notin \mathscr{X}'$ and $\exists \mathscr{Y} \subseteq \mathscr{X}' : \bigcup \mathscr{Y} = X_{\hat{k}} \wedge \tilde{c}(X_{\hat{k}}, \beta') < \sum_{X \in \mathscr{Y}} \tilde{c}(X, \beta')$ **then**
      Update $\mathscr{X}' \leftarrow (\mathscr{X}' \setminus \mathscr{Y}) \cup \{X_{\hat{k}}\}$;
    Remove $X_{\hat{k}}$ from $\mathscr{Z}$;
  if `iter` $= 1$ or $\sum_{X \in \mathscr{X}'} \tilde{c}(X) < \sum_{X \in \mathscr{X}} \tilde{c}(X)$ **then** $\mathscr{X} \leftarrow \mathscr{X}'$;
return $\mathscr{X}$;

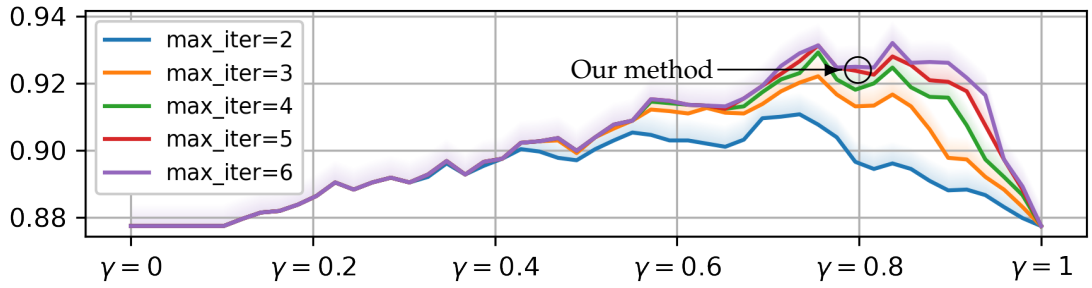


Figure 6.4. A posteriori assessment of global optimality of Algorithm 6.2 for different numbers of iterations (`max_iter`) and values of γ based on all instances of MinSetCover($\mathscr{S}$) in Algorithm 6.1 in our experiments. Each curve shows a *lower bound* (indicated by the shading) of the ratio of de-facto exact solutions.

MaxSetPacking$_{\text{LP}}$ serves the only purpose to determine whether computation of the exact set energy $\tilde{c}(X)$ in line 11 is necessary. Relaxing this bound thus, at worst, leads to more frequent computations of the exact set energy than necessary, but does not affect the segmentation result. This means that for MaxSetPacking$_{\text{LP}}$, fast run time is more important than accuracy. We thus approximated MaxSetPacking$_{\text{LP}}$ by packing disjoint elements of $\mathscr{S}$ in decreasing order of their respective set energies c and found that it overall yields the lowest run time.

6.3.4 Post-processing

Post-processing of the segmentation result (i.e. the globally optimal solution $\mathscr{X}$) is performed by refining the segmentation masks (e.g., hole filling using morphological operations) and rejecting falsely detected objects (e.g., imaging artifacts or debris objects). Details are given in Appendix C.4. Note that objects are neither split nor merged in the post-processing.

6.4 Evaluation

We have applied the proposed SuperDSM approach to 2-D fluorescence microscopy image data. For performance evaluation, we have used six image datasets of five different cell types comprising various challenges, which are described in Section 3.2 (NIH3T3, U2OS, GOWT1, Fibroblast, HeLa). The data is challenging for a variety of reasons, including autofluorescence artifacts, low signal-to-noise ratio, closely clustered and partially overlapping objects, strongly non-elliptical shapes, and different object scales. Overall, 348 images were used, including 5593 annotated cell nuclei in total.

We applied the SuperDSM approach using the *same set of hyperparameters* for all six datasets. In addition, we applied our approach using dataset-specific adaptations (SuperDSM*). Details on the hyperparameters are given in Appendix D.

6.4.1 Experimental results

We studied the segmentation accuracy and the cluster splitting performance, and carried out a quantitative comparison with previous methods. We used region-based and contour-based performance measures (SEG, Dice, Rand, HSD, NSD), as well as detection-based measures (Merge, Split), which are described in Section 3.4. We also provide an analysis of the run time performance and describe the application of our approach to another imaging modality.

We performed a comparison with previous methods, including eight state-of-the-art methods (CVX-LS, RPL, Blob-WS, Bayes-LS, CPN, SEG-SELF, RFOVE, Cellpose) comprising those which were reported to achieve the best results on the respective datasets and three standard methods (Otsu, Blob-LS, and Blob-RW). The methods

are described in Section 3.3. For the three standard methods, Gaussian filtering and morphological closing were applied for pre- and post-processing, and the hyperparameters were optimized for each dataset using a grid search scheme, which maximizes the average Dice and SEG values using two randomly chosen images per dataset. These images are also included in the test set. Note that we did not use grid search for our method, which is more realistic in practical applications. Also, we are on the safe side that we do not give an advantage to our method in the comparison. In addition, we included our previous method GOCELL (Chapter 5) in the comparison.

For the U2OS and NIH3T3 datasets, we computed the performance values for SEG-SELF and RFOVE based on the *original* segmentation results published by the authors. Some results for HSD and NSD are somewhat different from previously reported values (Section 5.4.3) since object correspondences were established differently (cf. Section 3.4). Results for HSD differ from [65, 66], possibly due to different computation of object correspondences. For the two GOWT1 datasets, we applied the original SEG-SELF and RFOVE implementations. In addition, we applied Cellpose to all six datasets. Note that the authors had used the NIH3T3 and U2OS datasets for training the network. Thus, the results of Cellpose for these two datasets should be treated with caution. For the NIH3T3 and the two GOWT1 datasets, we had to manually adapt one input parameter of Cellpose to achieve useful results (the optimal nuclei diameter, which we determined based on the ground truth, and which by default is computed automatically). For CVX-LS, RPL, Blob-WS, Bayes-LS, and CPN, we have used the performance values provided in the respective publications. For CPN, only performance values for the SEG measure were reported.

The results for all approaches and all datasets are provided in Table 6.1. Below, we discuss the datasets individually.

NIH3T3 dataset

The NIH3T3 dataset contains many closely clustered cell nuclei. SuperDSM yields better results than SEG-SELF and RFOVE regarding SEG, Dice, Rand, HSD, NSD, and Split. Compared to RPL, SuperDSM yields a slightly worse Dice value, but improved NSD and strongly improved HSD values. The lowest number of falsely merged/split objects is obtained by Cellpose and the second-lowest by GOCELL and SuperDSM. However, for Cellpose this dataset was used for training, and GOCELL employs dataset-specific parameters and pre-processing (see Sections 5.4.3–5.4.4). For SEG, Dice, HSD, and NSD, SuperDSM performs substantially better than Cellpose. Besides using fixed parameters for all datasets, we also employed dataset-specific adaptations for our approach (SuperDSM*) and discarded objects by post-processing which likely correspond to autofluorescence artifacts (based on the radius and connected component analysis). SuperDSM* yields the best results

	SEG	Dice	Rand	HSD	NSD	Merge	Split
NIH3T3 cells							
RPL	—	0.91	0.93	14.1	0.09	—	—
SEG-SELF	0.80	0.89	0.92	12.9	0.11	0.8	1.3
GOCELL	0.84	**0.92**	**0.94**	**8.3**	**0.06**	0.7	0.5
RFOVE	0.80	0.89	0.92	13.3	0.12	0.9	0.9
Cellpose	0.75	0.86	0.92	33.4	0.17	**0.0**	**0.4**
SuperDSM	0.82	0.90	0.93	8.8	0.08	0.8	**0.4**
SuperDSM*	**0.85**	**0.92**	**0.94**	**8.3**	0.07	0.5	0.6
U2OS cells							
CVX-LS	—	0.94	—	12.8	0.05	—	—
RPL	—	**0.96**	**0.96**	10	**0.02**	—	—
Bayes-LS	—	**0.96**	—	12.7	—	—	—
SEG-SELF	0.85	0.94	0.95	13.7	0.08	0.3	4.6
GOCELL	0.75	0.92	0.93	15.5	0.09	0.4	3.3
RFOVE	0.77	0.92	0.93	15.8	0.16	1.8	1.9
Cellpose	0.89	**0.96**	**0.96**	11.6	0.06	**0.1**	**0.2**
SuperDSM	0.86	0.93	0.94	8.8	0.05	0.6	0.5
SuperDSM*	**0.90**	**0.96**	**0.96**	**7.3**	0.06	0.9	0.4
GOWT1 dataset 1							
Blob-WS	0.74	—	—	—	—	—	—
CPN	0.85	—	—	—	—	—	—
SEG-SELF	0.52	0.89	0.97	30.3	0.17	**0.0**	0.1
GOCELL	0.85	**0.94**	**0.98**	4.7	0.02	**0.0**	**0.0**
RFOVE	0.60	0.89	0.97	25.6	0.15	**0.0**	0.1
Cellpose	0.72	0.86	0.97	28.2	0.16	0.1	**0.0**
SuperDSM	0.84	**0.94**	**0.98**	4.3	**0.01**	**0.0**	**0.0**
SuperDSM*	**0.87**	**0.94**	**0.98**	**4.2**	**0.01**	**0.0**	**0.0**
GOWT1 dataset 2							
Blob-WS	0.91	—	—	—	—	—	—
CPN	0.87	—	—	—	—	—	—
SEG-SELF	0.82	0.92	0.97	18.4	0.12	**0.0**	1.1
GOCELL	0.91	**0.95**	**0.98**	3.9	**0.01**	**0.0**	**0.0**
RFOVE	0.79	0.90	0.97	18.9	0.13	**0.0**	0.7
Cellpose	0.73	0.92	0.97	18.2	0.13	**0.0**	0.6
SuperDSM	0.89	0.94	**0.98**	4.3	0.02	**0.0**	0.5
SuperDSM*	**0.92**	0.94	**0.98**	**3.6**	**0.01**	**0.0**	**0.0**
Fibroblasts							
Otsu	0.78	0.86	0.97	12.4	0.14	0.1	0.9
Blob-LS	0.72	0.83	0.96	18.5	0.18	0.1	1.2
Blob-RW	0.29	0.64	0.93	36.9	0.28	**0.0**	0.1
GOCELL	0.93	**0.90**	**0.98**	6.5	**0.01**	**0.0**	**0.0**
Cellpose	0.54	0.56	0.94	115.0	0.39	**0.0**	0.3
SuperDSM	0.94	0.89	**0.98**	5.8	0.02	**0.0**	0.1
SuperDSM*	**0.95**	**0.90**	**0.98**	**5.1**	**0.01**	**0.0**	**0.0**
HeLa cells							
Otsu	0.85	**0.94**	**0.98**	**10.5**	0.08	0.2	2.8
Blob-LS	0.85	0.93	**0.98**	13.5	0.06	0.2	0.2
Blob-RW	0.68	0.81	0.94	31.7	0.15	**0.0**	0.4
GOCELL	0.89	**0.94**	**0.98**	15.9	0.03	**0.0**	0.3
Cellpose	0.69	0.76	0.95	106.7	0.24	**0.0**	0.3
SuperDSM	**0.90**	0.93	**0.98**	13.9	**0.02**	0.1	**0.0**
SuperDSM*	**0.90**	**0.94**	**0.98**	13.2	0.03	0.1	**0.0**

Table 6.1. Segmentation performance of different approaches. For SEG, Dice, Rand, higher is better. For HSD, NSD, Merge, Split, lower is better. Not available results are indicated by "—". Best results are highlighted.

for SEG, Dice, Rand, HSD, and Merge. An example segmentation result is shown in Figure 6.5. It can be seen that clustered and non-clustered cell nuclei are well segmented.

U2OS dataset

The U2OS dataset is difficult due to strongly non-elliptical shape of the cell nuclei, which is challenging for the merging/splitting schemes of the segmentation methods. Cellpose yields the lowest number of falsely merged/split objects, however, this dataset was used for training the network. SEG-SELF achieves the second-lowest false merging rate (0.3 per image), but has a strong tendency to over-segmentation (4.6 falsely split objects per image). SuperDSM yields only 0.6 falsely merged and only 0.5 falsely split cell nuclei per image. This good merging/splitting performance is in agreement with an improved SEG performance (SEG is sensitive to false merges/splits). Compared to RPL, SuperDSM performs worse for Dice, Rand, and NSD. However, Dice and Rand are invariant to false merges/splits, which were not reported for RPL. SuperDSM* yields the best results for SEG, Dice, Rand, and HSD. SEG-SELF and GOCELL yield fewer false merges, but significantly more false splits. Thus, overall, SuperDSM* performs best. Figure 6.6 shows example segmentation results. Our approach yields no false merges/splits, whereas SEG-SELF yields four falsely split cell nuclei. The object contours of all cell nuclei are accurately segmented.

GOWT1 datasets

For GOWT1 dataset 1, SuperDSM yields overall very good results. Compared to SEG-SELF and RFOVE, strong improvements can be observed for SEG (0.60 to 0.84), Dice (0.89 to 0.94), HSD (25.6 to 4.3), and NSD (0.15 to 0.01). Improvements are also large compared to Cellpose and concern mostly SEG (0.72 to 0.84), Dice (0.86 to 0.94), HSD (28.2 to 4.3), and NSD (0.16 to 0.01). GOCELL and CPN perform slightly better (SEG is 0.85), however, only SEG was reported for CPN (which is invariant to false-positive detections) and GOCELL used dataset-adapted parameters and pre-processing. Using our method with dataset-specific adaptations (SuperDSM*) yields the best results for all performance measures. For GOWT1 dataset 2, SuperDSM performs competitively. SEG is improved compared to CPN, SEG-SELF, RFOVE, and Cellpose, but not as good as Blob-WS (which did not perform well for GOWT1 dataset 1). SuperDSM* yields the best results regarding all measures. Figure 6.7 shows example segmentation results. All objects are accurately segmented, including the low-intensity cell nuclei and the irregularly shaped nucleus in GOWT1 dataset 2.

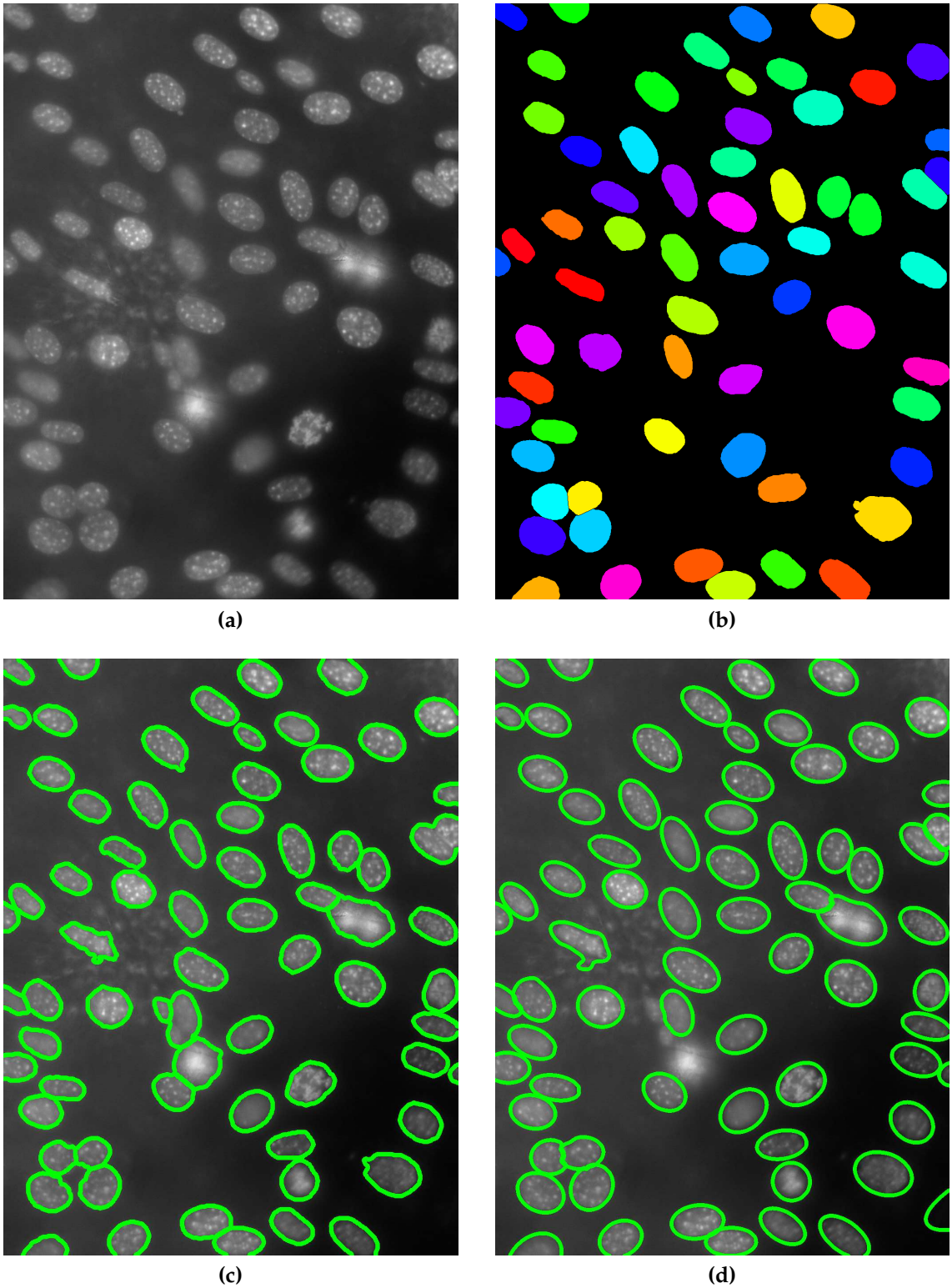


Figure 6.5. Example segmentation results (green contours) for the NIH3T3 dataset. (a) Original image. (b) Ground truth. (c) Result of RFOVE. (d) Result of SuperDSM*.

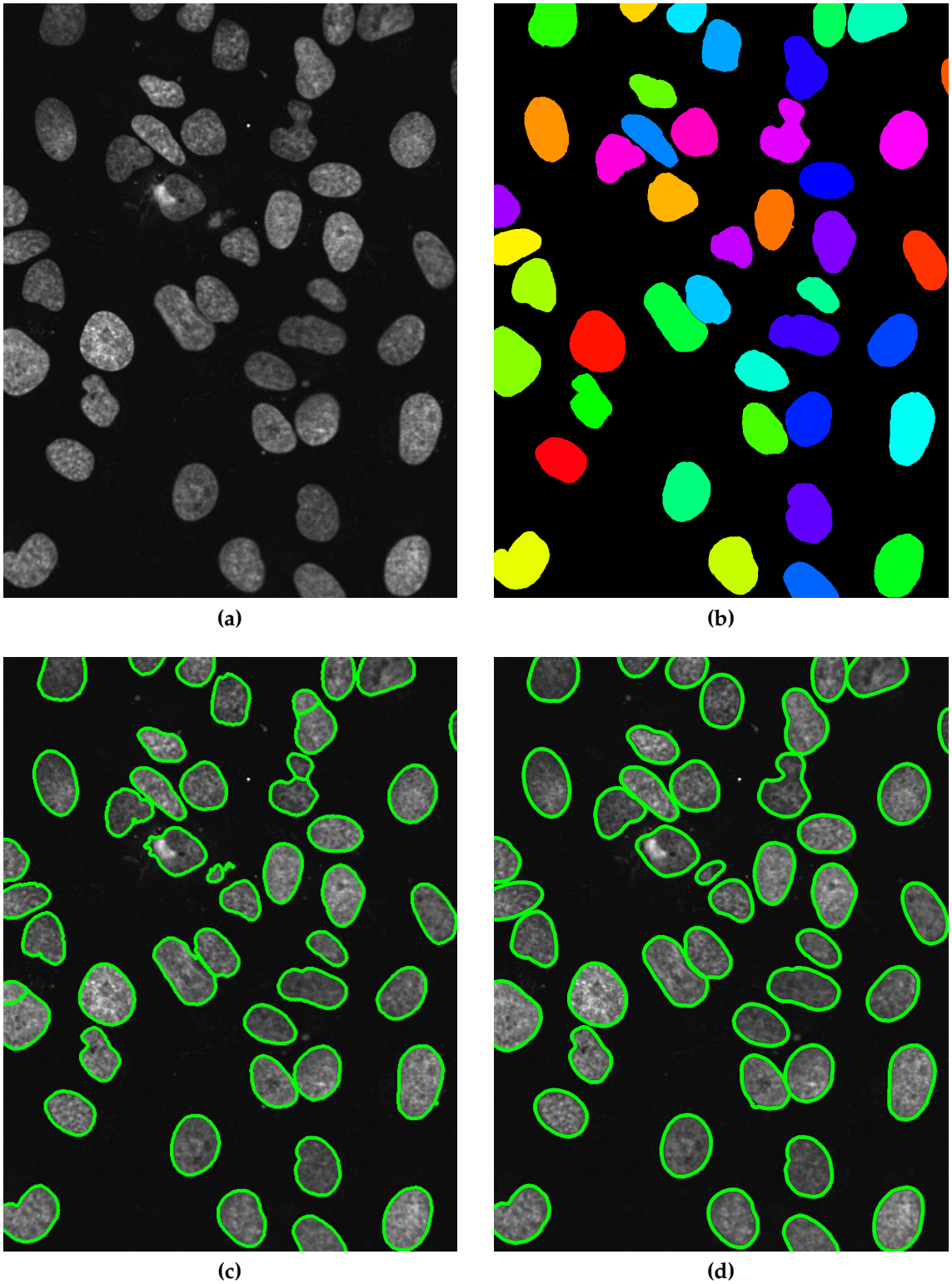


Figure 6.6. Example segmentation results (green contours) for the U2OS dataset. (a) Original image (contrast-enhanced). (b) Ground truth. (c) Result of SEG-SELF. (d) Result of SuperDSM*.

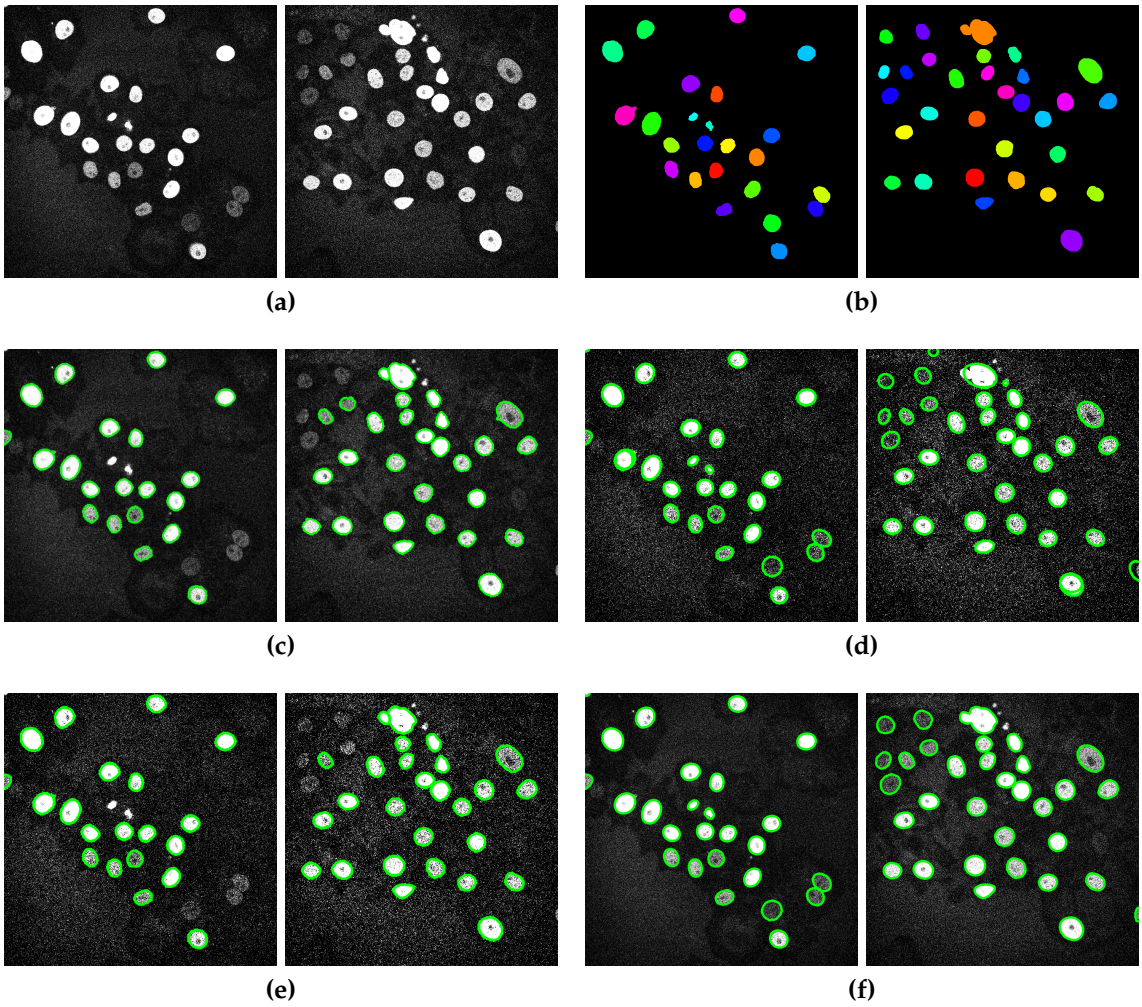


Figure 6.7. Example segmentation results (green contours) for GOWT1 dataset 1 (left) and GOWT1 dataset 2 (right). (a) Original images (contrast-enhanced). (b) Ground truth. (c) Result of SEG-SELF. (d) Result of GOCELL. (e) Result of RFOVE. (f) Result of SuperDSM*.

Fibroblast dataset

For the Fibroblast dataset, SuperDSM yields the best results for SEG, Rand, HSD, and Merge, but is slightly worse for Dice, NSD, Split compared to GOCELL. Compared to Cellpose, the results are strongly improved for SEG (0.54 to 0.94), Dice (0.56 to 0.89), Rand (0.94 to 0.98), HSD (115.0 to 5.8), NSD (0.39 to 0.02), and Split (0.3 to 0.1). The dataset contains multiple images with only few cell nuclei and it turns out that the pre-processing somewhat underestimates the scale σ. Adapting this parameter (SuperDSM*) yields the best results for all measures. Example segmentation results are shown in Figure 6.8. Our approach reliably segments the contours of both elliptical and non-elliptical cell nuclei. GOCELL performs worse since elliptical shape models are insufficient. Blob-LS and Blob-RW perform worse since their initialization is prone to the non-elliptical shapes and closely clustered objects.

HeLa dataset

For the HeLa dataset, SuperDSM performs overall better than Otsu, Blob-LS, Blob-RW, and Cellpose. Compared to GOCELL, SuperDSM yields better results for SEG, HSD, NSD, and Split. Overall, SuperDSM yields the best result for SEG, Rand, NSD, and Split. SuperDSM* yields the overall best result.

Conclusion

Considering all datasets, it turns out that our SuperDSM approach achieves better results than previous methods for the Fibroblast and HeLa datasets and competitive results for the U2OS, NIH3T3, GOWT1 datasets using the same set of hyperparameters for all six datasets. The results are generally slightly worse than using dataset-specific adaptations (SuperDSM*). Using such adaptations, our approach generally yields best results for all datasets for the region-based measure SEG. For Dice and Rand, the results are competitive. Concerning contour-based measures, the performance of our method is best for four out of six datasets, and second-best for two datasets. Regarding cluster splitting, our method generally yields the best results for all datasets for the number of falsely merged/split objects (sum of Merge and Split in Table 6.1), and achieves very low Merge and Split counts (less than one falsely merged/split objects per image). For the NIH3T3 and U2OS datasets, Cellpose yields fewer falsely merged/split objects, however, both datasets were used for training the network. For the other datasets, our method achieves strongly improved results compared to Cellpose for all region-based and contour-based measures. The best performing methods besides our approach are RPL and GOCELL. However, for RPL, published results for SEG, Merge, and Split are not available. GOCELL yields worse results than SuperDSM for non-elliptical cell nuclei (e.g., U2OS dataset). We thus conclude that our method performs overall best in this study.

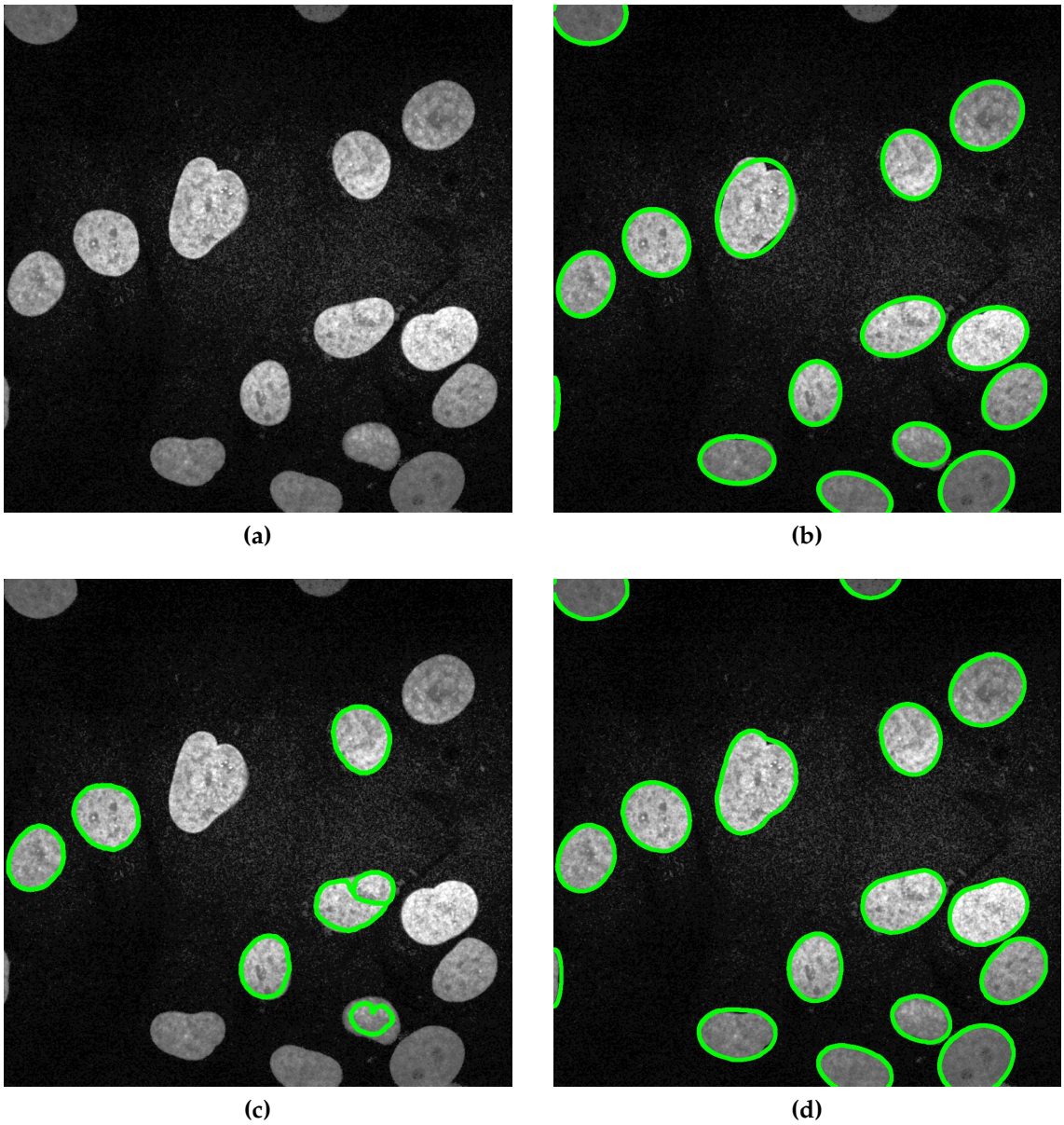


Figure 6.8. Example segmentation results (green contours) for the Fibroblast dataset. (a) Original image (contrast-enhanced). (b) Result of GOCELL. (c) Result of Cellpose. (d) Result of SuperDSM*.

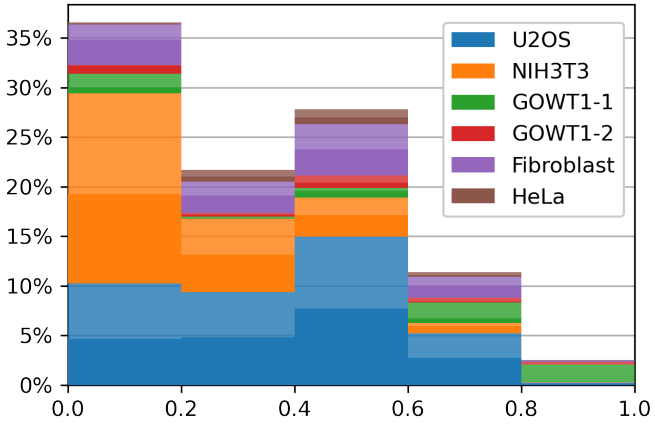
In practical applications, the hyperparameters of the global energy minimization of our SuperDSM* method can be adapted as follows starting from the default values (SuperDSM). For example, when the computed cell contours are too smooth, one could reduce the weight α of the regularization of the deformations in Eq. (6.7). When objects are falsely split (e.g., due to irregular shapes) one could increase the constant term β of the extended set energy functions in Eq. (6.15). The scale σ can be adapted by using $\sigma = R/\sqrt{2}$, where R is the average radius of cell nuclei which could be determined coarsely.

6.4.2 Run time performance

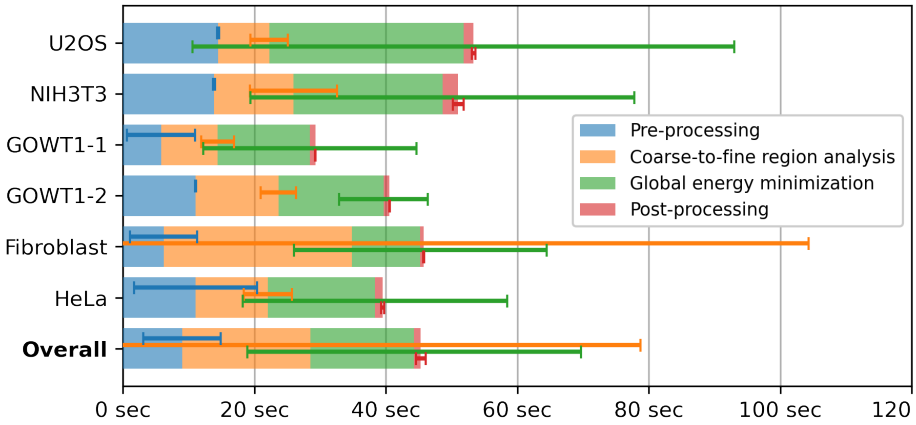
In addition, we studied the run time performance of our approach. As described in Section 6.3, our approach separately processes regions of possibly clustered objects. This involves the computation of the extended set energies $\tilde{c}(X)$ in Eq. (6.15) for different sets $X \in \mathbb{P}(U)$ to determine the solution $\mathscr{X} \subseteq \mathbb{P}(U)$ of $\textsc{MinSetCover}(\mathbb{P}(U))$, see Algorithm 6.1. The computational cost is the number of sets (image regions), for which the set energies must be computed (see Section 6.2.5). This number is upper-bounded by the cardinality of the universe of a region of possibly clustered objects. Regions of possibly clustered objects with universe cardinality $\#U \leq 2$ are computationally cheap, since $\#\mathbb{P}(U) \leq 3$ and thus at most three sets must be computed. We found that such regions occurred in 82.7 % of the cases. The other cases are computationally more challenging, and our method reduces the computational cost by excluding sets *before* computing the corresponding energies using Criterion 6.1 and Criterion 6.2. The more sets are excluded, the more efficient the method is. As a measure of efficiency, we use the *set exclusion rate (SER)*, which is the ratio of excluded sets compared to the cardinality of $\mathbb{P}(U)$. To quantify the computational cost of our approach for the computationally challenging cases, below we consider SER for regions of possibly clustered objects with universe cardinality $\#U > 2$. We also study the run time of our method.

Figure 6.9a shows a histogram of SER values for the computationally challenging cases ($\#U > 2$) for all six datasets. The median SER is 0.33 and the maximum is 0.97, thus, the computational cost was typically reduced by 33 % and at best by 97 %. For the U2OS, GOWT1, Fibroblast, and HeLa datasets, relatively high SER values were obtained. The median SER is 0.62 for GOWT1 dataset 1 and 0.43 for the other datasets. Thus, the computational cost was typically reduced by 43 % or 62 %, respectively. Only for the NIH3T3 dataset lower SER values were obtained. The reason is probably that this dataset is most difficult in terms of clustered objects. Overall, we can conclude that the two criteria effectively reduce the computational cost for the computationally challenging cases.

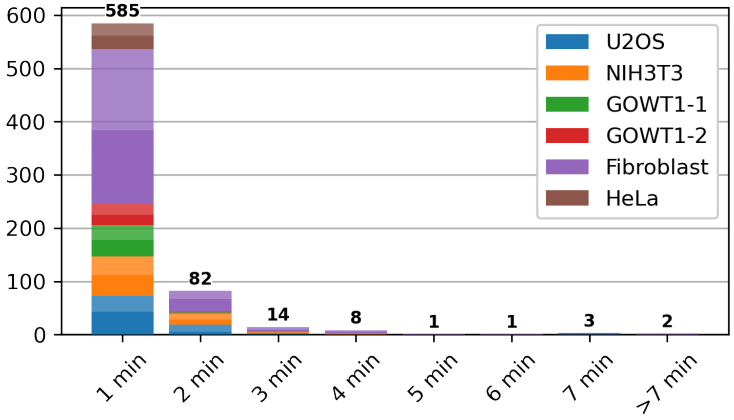
Figure 6.9b shows the average computation time per image of the individual processing steps of our approach. It can be seen that both pre-processing and post-processing performed in almost constant time (see the scattering indicated by the



(a)



(b)



(c)

Figure 6.9. Run time performance of our approach for the six datasets (darker color shades correspond to SuperDSM, brighter color shades correspond to SuperDSM*). (a) Histogram of the set exclusion rate SER (relative frequencies). (b) Computation time of the individual processing steps of our approach (mean and standard deviation). (c) Histogram of the total run time per image (absolute frequencies). The labels on the horizontal axis denote intervals (e.g., 0 to 1 min, 1 to 2 min).

error bars). The computationally most expensive task in pre-processing is scale estimation (cf. Section 6.3.1). The strongest scattering in the average run time can be observed for the coarse-to-fine region analysis, which is probably due to the linear search to determine seeds for splitting regions with high normalized energies (cf. Section 6.3.2). Regarding the global energy minimization (cf. Section 6.3.3), we observe that images with many closely clustered objects correlate with longer run times (U2OS and NIH3T3 datasets). This confirms that our approach efficiently copes with non-clustered cell nuclei. The overall average run time was 45.3 seconds per image. For comparison, the average run time of our previous globally optimal GOCELL approach (Chapter 5) was 1:23 minutes per image using the same hardware (see below). Thus, we achieved a speed-up of 183 %. This is remarkable, since deformable shape models used in the proposed approach are computationally much more challenging than elliptical models due to the higher dimension of the parameter space. For the globally optimal approach using circular models [64], we observed a run time of 38 minutes (Section 5.4.1). This emphasizes the computational efficiency of the proposed approach. In biological applications, most time is required for preparation of the specimen such as staining and for image acquisition (e.g., DAPI staining takes at least 2–4 minutes [6]). Thus, the run time of our method is suited for practical applications.

Figure 6.9c shows the overall run time of our method per image. From the mode of this histogram it can be seen that for the vast majority of the images, the typical computation time was less than 1 minute per image (84.1 % of the images). For 95.8 % of the images, the overall run time was less than 2 minutes. We observed only two cases, where the computation lasted unexpectedly long (a single image from the Fibroblast dataset, which took 14 minutes for SuperDSM and 12 minutes for SuperDSM*). The reason is probably that scale estimation was not accurate.

All experiments were performed using an AMD Ryzen Threadripper 3970X CPU and 32 GB of RAM. We used Intel Math Kernel Library 20.0 for fast sparse and dense linear algebra. For the coarse-to-fine region analysis, up to 16 regions of possibly clustered objects were processed in parallel. For global energy minimization, the energies for up to 16 sets were computed in parallel. For post-processing, up to 16 objects were processed in parallel. Note that no GPU acceleration was used for the experiments. Faster run times can be achieved by increasing the degree of parallelization, which is straightforward.

6.4.3 Application to different imaging modality

So far, we have studied the performance of our method using a wide range of fluorescence microscopy image data. In a final experiment, we investigated the applicability of our approach to another imaging modality, namely histopathology images stained with haematoxylin and eosin (H&E). These images are color images, where cell nuclei appear in blue or dark purple. We used the training dataset of the

MICCAI 2018 MoNuSeg challenge [125]. The image size is 1000×1000 pixels and the data contains 1390 cell nuclei from the histological section of a human stomach in adenocarcinoma disease condition (see Figure 6.10a). The data is challenging due to very densely clustered cell nuclei, a wide variety of nuclei shapes, and strongly inhomogeneous background.

We used minor methodological adaptations to account for the very different imaging modality. This concerns only pre-processing and the computation of Y_Ω (image intensities with τ_x offset). The idea is to transform the image intensities so that cell nuclei (dark purple regions) correspond to bright intensities. To this end, we first average the image intensities over the three color channels $g_x^{\mathrm{r}}, g_x^{\mathrm{g}}, g_x^{\mathrm{b}}$ using $g_x = 1 - \frac{1}{3}\left(g_x^{\mathrm{r}} + g_x^{\mathrm{g}} + g_x^{\mathrm{b}}\right)$. Second, we apply Gaussian filtering (standard deviation σ) followed by local maximum filtering ($2\sigma \times 2\sigma$ neighborhood) to determine the locally maximal responses $g_x^{\max}$. Third, we compute Y_Ω (image intensities with τ_x offset, see Eq. (6.5)) by $\tau_x = \max\left\{g_x^{\max}, \mathrm{mean}_{x\in\Omega}\, g_x\right\}$ using the mean intensity over all image points.

Our method performed the segmentation within 3:06 minutes and the result is shown in Figure 6.10c. We found that 97.1 % of the cell nuclei were detected (using the detection measure described in Section 3.4), and only 4.5 % were falsely merged or split, respectively. Given that our approach is designed for fluorescence microscopy images rather than H&E-stained histopathology images, the result is promising. In Figure 6.10c, right bottom, few small tissue regions are segmented since they are slightly darker than their neighborhood (as for cells). The result could be improved by not only taking into account the brightness, but also the color hue and saturation for computing Y_Ω. Overall, this experiment shows that our approach can be generalized to other imaging modalities.

6.5 Discussion

We have introduced a new *globally optimal* approach based on *deformable* shape models and global energy minimization for cell nuclei segmentation in microscopy images. The approach intrinsically copes with non-elliptical shapes, jointly exploits shape and intensity information, and is based on an *implicit* parameterization, which leads to a *convex* energy. Thus, energy minimization is independent of the initialization, fast, and robust. To jointly perform cell segmentation and cluster splitting, we have considered the infimum of the convex energy as a *set energy function*, i.e. a function of the set of image regions where model fitting is performed. We have proposed a novel iterative global energy minimization method, which provably determines the optimal image regions close to global optimality. The method exploits the inherent property of *superadditivity* of the set energy function, which is established via the *set-packing polytope*. Intuitively, the property of superadditivity means that a deformable shape model cannot fit better to an image region

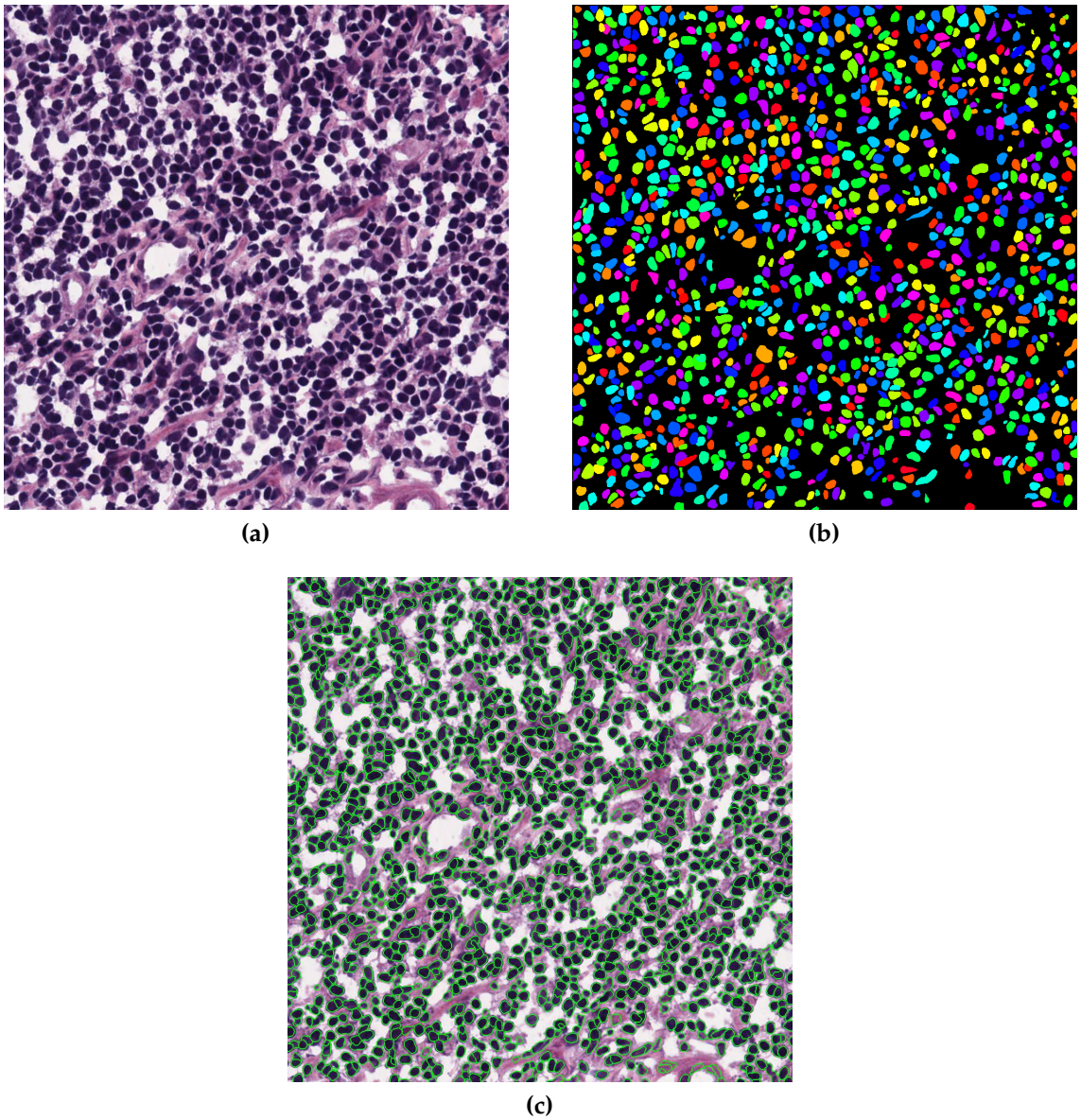


Figure 6.10. Application of our approach to different imaging modality. (a) Original H&E-stained histopathology image. (b) Ground truth. (c) Segmentation result (green contours) of SuperDSM*.

than it fits to any of its sub-regions. We have used this property to obtain a *necessary optimality condition* for the image regions, which shows that it is not required to consider all possible image regions for optimization. Instead, the proposed energy minimization method considers image regions in order of increasing size and leverages superadditivity to exclude regions corresponding to falsely merged objects using a fine-to-coarse scheme. This is achieved by iterative evaluation and refinement of the necessary optimality condition, and improves the computational efficiency, since when excluding a region, all its supersets are also excluded. We have also described a coarse-to-fine region analysis scheme, which determines the universe of region fragments used as input for global energy minimization. In addition, we have derived a closed-form solution of the proposed global energy minimization based on the superadditivity property for non-clustered cell nuclei, which further accelerates the computation.

The regularization parameter α of the convex energy is used to control the shape variability of the deformable shape models. An *extended set energy function* has been introduced to avoid over-segmentation, which uses the hyperparameter β defining the maximum allowed energy difference for merging two image regions (i.e. two deformable shape models that are fitted in these regions). Our approach automatically determines scale-related hyperparameters based on scale estimation. The objective function of our global energy minimization method corresponds to a min-weight set-cover problem, which is **NP**-hard to compute. We have thus used a fast approximation algorithm, which determines a solution close to global optimality. In addition, the design of the algorithm directly addresses the false splits and false merges possibly introduced by using an approximation. We have performed an analysis of global optimality and found that the global solution was exactly determined in at least 92.1 % of our experiments, the average approximation ratio of the solution was at least 99.7 %, and the median was 100.0 %. To compute the set energy function, we have used a fast numerical second-order method which directly determines global solutions by convex energy minimization.

We have applied our approach to a wide range of 348 fluorescence microscopy images of five different cell types comprising 5593 cell nuclei, and performed a quantitative comparison with previous methods. It turned out that our approach generally demonstrates the best or second-best cluster splitting performance. The segmentation accuracy is better compared to previous methods according to region-based measures, and is competitive according to contour-based measures. For the region-based SEG performance measure used in the cell segmentation benchmark [104], which is the best suited measure for overall segmentation performance (since it incorporates both detection and object-based segmentation performance), our approach generally yields superior results for all datasets. Our approach is robust since it achieves competitive or improved results even when using a fixed set of hyperparameters for all datasets, compared to nine state-of-the-art methods which previously achieved best results on the respective datasets. In addition, we have demonstrated that our approach can be generalized to other imaging modalities.

Chapter 7

Summary and outlook

7.1 Summary

In this thesis, three new globally optimal model-based approaches for cell nuclei segmentation have been introduced. The new approaches address main challenges and difficulties of cell segmentation in general and for fluorescence microscopy in particular (see Section 1.2), and can be summarized as follows:

- In fluorescence microscopy imaging, strong image noise occurs due to low intensity of light emitted by the fluorophores. To cope with this, we have introduced the CVXELL approach (Chapter 4). The approach is based on convex optimization and elliptical models, which are fitted directly to the image data, using a sequential approximation scheme. Convex optimization has the advantage that globally optimal solutions are determined independently of the initialization.

- To better cope with touching and partially overlapping cell nuclei, we have proposed the GOCELL approach (Chapter 5). The approach is based on elliptical models and a multi-object scheme, which *jointly* performs cell segmentation and cluster splitting, meaning that neither prior object detection nor prior image binarization is required. Global energy minimization is performed using an efficient combination of convex and combinatorial optimization, which determines the solution close to global optimality. GOCELL is also intrinsically robust to inhomogeneous image intensities.

- We have also introduced the SuperDSM approach, which is based on *deformable* shape models (Chapter 6). The approach *jointly* performs *globally optimal* cell segmentation and cluster splitting, but in contrast to the GOCELL approach, SuperDSM is *scale-invariant* and more general, since it naturally copes with cell nuclei of varying shapes. Despite its higher level of generality, SuperDSM is also more efficient since it exploits the inherent property of *superadditivity* for combinatorial optimization.

The proposed approaches were quantitatively evaluated using publicly available benchmark datasets, comprising different challenges, cell types, and stainings. The proposed approaches generally achieved state-of-the-art or improved performance. In particular, SuperDSM achieved competitive results even when using a fixed set of hyperparameters for all datasets.

Our main technical contributions and findings can be summarized as follows:

- We have introduced implicitly parameterized shape models for cell nuclei segmentation. We have also described model fitting methods based on energy minimization and proposed three different parameterizations. In the first method (CVXELL), an *elliptical* model is parameterized as a function of the center of the model. This non-linear parameterization has the advantage that the center of the model can be used to encode prior knowledge regarding the location of the model during energy minimization (e.g., for priorly detected objects). Energy minimization using this parameterization is performed by *convex* optimization based on a sequential approximation scheme. In the second method (GOCELL), a *linear* parameterization for elliptical models was introduced. This parameterization has the advantage that it yields a *convex* energy function, so that direct minimization can be performed without requiring approximation. Another advantage is that the energy function is invariant to the image contrast. In the third method (SuperDSM), we proposed a *linear* parameterization for *deformable* shape models. This parameterization also leads to a *convex* energy function for model fitting, but permits more general shapes than previous parameterizations.

- The image intensities are directly encoded in the energy functions. Model fitting using parametric shape models thus *jointly* exploits shape and intensity information. The proposed energy minimization methods are based on *convex* optimization. We employed fast *second-order* optimization methods for convex optimization, which are robust and yield globally optimal solutions independently of the initialization. Convex optimization exploiting shape information was not used in previous cell segmentation methods.

- Single-object and multi-object schemes have been proposed. *Single-object* schemes have the advantage, that model fitting can be performed by energy minimization using exclusively *convex* optimization methods (e.g., Chapter 4, Section 5.2, Section 6.2.2). However, prior object detection is required since cell microscopy images generally contain multiple objects. To avoid prior object detection, we have also proposed *multi-object* schemes (e.g., Section 5.3, Section 6.2.3). In Chapter 5, we generalized the single-object model to the multi-object case. The multi-object model yields a non-convex energy, yet we have found that model fitting using the multi-object model corresponds to the *min-weight set-cover* problem. In this formalism, an overcomplete set of *region prototypes* is used and each region prototype is weighted by a

non-negative energy value determined by convex optimization. Joint cell segmentation and cluster splitting is performed by determining a minimum-weight subset of region prototypes. We then used this result to design a global multi-object energy minimization scheme, where the overcomplete set of region prototypes is computed in advance (Section 5.3). In Chapter 6, we proposed a more efficient multi-object energy minimization scheme, which iteratively evaluates and refines a *necessary optimality condition* for the region prototypes (see below). This scheme automatically confines the computations to a meaningful subset of region prototypes (instead of computing all possible region prototypes in advance).

- We have found a *necessary optimality condition* for region prototypes, which is based on the property of *superadditivity*. This property means that the weight of any region prototype is lower-bounded by the sum of the weights of any pair of disjoint sub-regions and is established via the *set-packing polytope*. The advantage is that the computations are automatically confined to a meaningful subset of all region prototypes.

- We have also derived a *closed-form solution* of the proposed global energy minimization, using the corresponding min-weight set-cover and the property of superadditivity. The closed-form solution *directly* identifies non-clustered cell nuclei and determines the corresponding segmentation result, instead of performing cell cluster splitting, which further accelerates the computations.

- Computing the min-weight set-cover is **NP**-hard and thus computationally intractable. Hence, we used fast approximation algorithms, which directly address the false splits and false merges possibly introduced by the approximation and are *guaranteed* to determine a solution close to global optimality. In addition, global optimality was checked a posteriori and we found, that the global solution was exactly determined in at least 91.5 % (GOCELL) and 92.1 % (SuperDSM) of the experiments, respectively.

- We have used *region fragments* to characterize the overcomplete set of region prototypes as a set of a computationally tractable cardinality. In addition, we proposed a *coarse-to-fine region analysis* scheme, which determines the region fragments by recursively fitting elliptical models using convex optimization.

- The proposed segmentation approaches were designed for cell segmentation in fluorescence microscopy images. In addition, we have demonstrated the general applicability of SuperDSM to other imaging modalities (H&E stained histopathology images).

7.2 Outlook

In future work, the proposed approaches could be extended as follows:

- The global optimization scheme of the SuperDSM approach is based on a *necessary* optimality condition for the region prototypes, which is evaluated and refined iteratively. To increase the computational efficiency of the global optimization scheme, an extension would be to also incorporate *sufficient* optimality conditions into the iterative scheme.
- The proposed global energy minimization schemes minimize a non-convex multi-object energy using convex and combinatorial optimization steps, which are performed either successively (GOCELL) or alternatingly (SuperDSM). Blending both optimization steps into a single optimization method might further increase efficiency and accuracy.
- We have shown that SuperDSM can be used for segmentation of cell nuclei in microscopy data of different imaging modalities. Future research will consider extendability to other imaging modalities and other application areas of computer vision and image analysis.

Appendix

In this appendix, we provide mathematical, algorithmic, and implementation details of Chapter 6. In particular, we provide details on the computation of the image intensity offsets (Appendix A), mathematical details and proofs on convex optimization and superadditivity (Appendix B), algorithmic and implementation details of Algorithm 6.1, coarse-to-fine region analysis, convex optimization, and post-processing (Appendix C), and the hyperparameters (Appendix D).

A Computation of the image intensity offsets

Notation. We use g_x to denote the image intensity at image point $x \in \Omega$. By stacking the image intensities in an arbitrary but fixed order $\Omega = \{x^{(1)}, \ldots, x^{(\#\Omega)}\}$, we obtain a vector representation $g = (g_{x^{(1)}}, \ldots, g_{x^{(\#\Omega)}})$ of the whole 2-D image. Accordingly, given a vector w, we write w_x to denote the component of the vector w corresponding to the image point x. Using this vector notation, let $\mathcal{G}_\sigma$ be the vector representation of a 2-D Gaussian function with standard deviation σ, and let "$*$" denote the corresponding 2-D convolution. Thus, for example, $(\mathcal{G}_\sigma * g)_x$ denotes the intensity value of the Gaussian-filtered image g at image point x.

As described in Section 6.2.2, the purpose of the image intensity offsets $\tau_x \in \mathbb{R}$ is to *roughly* subdivide an image into two regions which are foreground ($g_x > \tau_x$) or background ($g_x < \tau_x$). To determine the intensity offsets, one possibility would be to use $\tau_x = (\mathcal{G}_\sigma * g)_x$. However, this leads to a misclassification of low-intensity objects as image background ($g_x - \tau_x < 0$) if they are close to bright objects (e.g., see the object in the center in Figure A.1b). To better cope with such intensity inhomogeneities, we propose the following two-step scheme. First, we determine the image region $\Omega' = \{x \in \Omega | g_x \leq \tau_{\max}\}$ using $\tau_{\max} = 3 \cdot \operatorname{std}_{x \in \Omega} g_x$, where $\operatorname{std}_{x \in \Omega} g_x$ is the standard deviation of the image intensities g_x for $x \in \Omega$. Second, we compute the intensity offsets τ_x either *(i)* directly using $(\mathcal{G}_\sigma * g)_x$ for image points $x \in \Omega'$, or *(ii)* by clipping image intensities higher than $\tau_{\max}$ prior to filtering. For image points $x \in \Omega'$ close to the boundary of Ω', the intensity offsets are interpolated

between these two options using the weights λ_x. Formally, this corresponds to

$$\tau_x = \lambda_x \cdot (\mathcal{G}_\sigma * g)_x + (1 - \lambda_x) \cdot (\mathcal{G}_\sigma * \mathrm{clip}\,(g, \tau_{\max}))_x\,,$$
$$\text{where } \lambda_x = \begin{cases} 1 & \text{if } x \in \Omega' \\ \max\left\{0, 1 - \mathrm{dist}_{\Omega'}(x)/\sigma\right\}^2 & \text{else,} \end{cases}, \tag{A.1}$$

where $\mathrm{clip}\,(g, \tau_{\max})$ denotes clipping of the image intensities and $\mathrm{dist}_{\Omega'}(x) = \min_{x' \in \Omega'} \|x - x'\|_2$ is the Euclidean distance of the image point x to the set Ω'. It can be seen in Figure A.1c, that using Eq. (A.1) properly classifies all three objects as image foreground ($g_x - \tau_x > 0$).

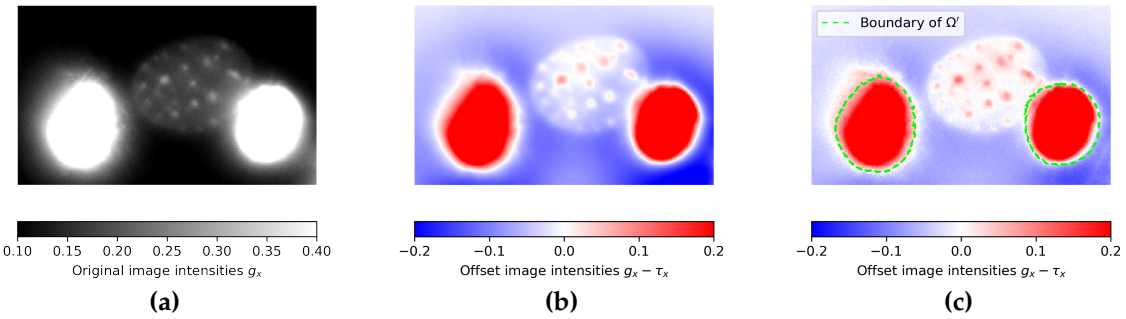


Figure A.1. Computation of image intensity offsets. (a) Original image section (NIH3T3 cells, contrast-enhanced to improve visibility). (b) Image intensities $g_x - \tau_x$ using the offsets $\tau_x = (\mathcal{G}_\sigma * g)_x$. (c) Image intensities $g_x - \tau_x$ using the offsets computed by Eq. (A.1). The dashed contour corresponds to the boundary of Ω'.

B Proofs and mathematical details

B.1 Convexity and smooth approximation

A direct numerical solution of Eq. (6.7) is difficult due to the non-smooth regularization term $\|\xi\|_1$. We thus propose using the smooth approximation

$$\|\xi\|_1 \approx \left\langle \mathbb{1}_{\#\Omega}, \sqrt{\xi^2 + \epsilon} \right\rangle - \sqrt{\epsilon} \cdot \#\Omega, \tag{B.1}$$

with $\epsilon > 0$, where the power of two and the square root are defined componentwise. Below, Property B.1 establishes the convexity of Eq. (6.7) using Eq. (B.1). In Eq. (B.1), the constant term $\sqrt{\epsilon} \cdot \#\Omega$ ensures that the approximation yields 0 for $\xi = \mathbb{0}$, which is important to retain Property B.2 (discussed thereafter).

Property B.1. *Eq.* (6.7) *states an unconstrained convex problem.*

Proof. *The absence of constraints is evident. To proof convexity of the objective function, let $\odot$ denote the Hadamard product and consider the Hessian matrix:*

$$\nabla^2 \psi_\omega(\theta, \xi) = \begin{bmatrix} F_\omega \\ G_\omega^\top \end{bmatrix} \cdot M \cdot \begin{bmatrix} F_\omega^\top & G_\omega \end{bmatrix} + \alpha \cdot \operatorname{Diag}\left(\mathbb{0}_6, \epsilon / \left(\xi^2 + \epsilon\right)^{3/2}\right), \tag{B.2a}$$

where

$$M = \operatorname{Diag}\left(Y_\omega^2 \odot \left(\zeta_\omega(\theta, \xi) - \zeta_\omega^2(\theta, \xi)\right)\right), \tag{B.2b}$$

$$\zeta_\omega(\theta, \xi) = \frac{1}{1 + \exp\left(Y_\omega \odot S_\omega(\theta, \xi)\right)}. \tag{B.2c}$$

Note that $M \succ 0$ due to $0 < \zeta_\omega(\theta, \xi) < 1$ for all θ, ξ. Thus, the Hessian matrix is of the form $\nabla^2 \psi_\omega(\theta, \xi) = H^\top H + \alpha D$, where (skipping the dependence of the matrices H and D on ω, ξ, θ for clarity of notation)

$$H = \operatorname{Diag} \sqrt{Y_\omega^2 \odot \left(\zeta_\omega(\theta, \xi) - \zeta_\omega^2(\theta, \xi)\right)} \cdot \begin{bmatrix} F_\omega^\top & G_\omega \end{bmatrix},$$
$$D = \operatorname{Diag}\left(\mathbb{0}_6, \epsilon / \left(\xi^2 + \epsilon\right)^{3/2}\right).$$

The matrix $H^\top H$ is positive semidefinite (PSD)*, since $z^\top H^\top H z = \tilde{z}^\top \tilde{z} \geq 0$ for all $z \in \mathbb{R}^{6+\#\Omega}$, where $\tilde{z} = Hz$ (since H is real-valued). The matrix D is also PSD due to $\epsilon > 0$. This means that the Hessian matrix $\nabla^2 \psi_\omega(\theta, \xi)$ is the sum of two PSD matrices, and thus also PSD. By the necessary and sufficient second-order condition for convexity (Section 2.2.2), the energy in Eq.* (6.7) *thus is a convex function.* □

B.2 Set-packing polytope

In Section 6.2.3, we define the set-packing polytope P for a given family of sets $X_1, \ldots, X_m \subseteq U$ as the set of solutions $\eta \in \mathbb{R}_+^m$ for the inequality in Eq. (6.13). Here, we consider the set-packing polytope P for a family of sets $\omega_1, \ldots, \omega_m \subseteq \Omega$, which is analogously defined as the set of solutions $\eta \in \mathbb{R}_+^m$ for the inequality

$$\sum_{k \in [m]} [x \in \omega_k] \cdot \eta_k \leq 1 \quad \text{for all } x \in \Omega. \tag{B.3}$$

An important property of the objective function in Eq. (6.7) is that $\psi_{\omega_1 \cup \cdots \cup \omega_m}(\theta, \xi)$ is an upper bound of $\eta_1 \cdot \psi_{\omega_1}(\theta, \xi) + \cdots + \eta_m \cdot \psi_{\omega_m}(\theta, \xi)$ for *any* pair of parameters θ, ξ and family of sets $\omega_1, \ldots, \omega_m$ with associated weights $\eta \in P(\omega_1, \ldots, \omega_m)$:

Property B.2. *Let $\omega_1, \ldots, \omega_m \subset \Omega$, $\omega = \omega_1 \cup \cdots \cup \omega_m$, and $\eta \in P(\omega_1, \ldots, \omega_m)$. Then*

$$\psi_\omega(\theta, \xi) \geq \eta_1 \cdot \psi_{\omega_1}(\theta, \xi) + \cdots + \eta_m \cdot \psi_{\omega_m}(\theta, \xi). \tag{B.4}$$

Proof. *Let ξ_x the component of ξ corresponding to $x \in \Omega$. Expanding Eq.* (6.7) *using Eq.* (6.8) *then yields*

$$\psi_\omega(\theta, \xi) = \sum_{x \in \omega} \ell_x(\theta, \xi) + \alpha \cdot |\xi_x|, \tag{B.5a}$$

where $\ell_x(\theta, \xi) = \ln\left(1 + \exp\left(-Y_\omega \cdot S_\omega(\theta, \xi)|_{\omega=\{x\}}\right)\right)$. Let $\chi_x = \sum_{k \in [m]} [x \in \omega_k] \cdot \eta_k$. Since $\chi_x \leq 1$ for all $x \in \omega$ due to $\eta \in P(\omega_1, \ldots, \omega_m)$,

$$\begin{aligned}
\psi_\omega(\theta, \xi) &\geq \sum_{x \in \omega} \chi_x \cdot (\ell_x(\theta, \xi) + \alpha \cdot |\xi_x|) && \text{(B.5b)}\\
&= \sum_{k \in [m]} \sum_{x \in \omega_k} \eta_k \cdot (\ell_x(\theta, \xi) + \alpha \cdot |\xi_x|) && \text{(B.5c)}\\
&= \sum_{k \in [m]} \eta_k \cdot \sum_{x \in \omega_k} \ell_x(\theta, \xi) + \alpha \cdot |\xi_x|, && \text{(B.5d)}
\end{aligned}$$

which directly yields Eq. (B.4). □

We use Property B.2 to proof Property B.2:

Property B.3 (L). *t $X_1, \ldots, X_m \subset U$, $X_k \neq \emptyset$ for all $k \in [m]$, and $\eta \in P(X_1, \ldots, X_m)$. Then, $\eta_1 \cdot c(X_1) + \cdots + \eta_m \cdot c(X_m) \leq c(X_1 \cup \cdots \cup X_m)$.*

Proof. *Let $h_X(\gamma) = \psi_{\tilde{\omega}(X)}(\gamma)$ and $\Gamma = \mathbb{R}^6 \times \mathbb{R}^{\#\Omega}$. The assumption $\eta \in P(X_1, \ldots, X_m)$ implies that*

$$\sum_{k \in [m]} [x \in \tilde{\omega}(X_k)] \cdot \eta_k \leq 1 \quad \text{for all } x \in \tilde{\omega}(X), \tag{B.6a}$$

where $X = X_1 \cup \cdots \cup X_m$, which means that $\eta \in P(\tilde{\omega}(X_1), \ldots, \tilde{\omega}(X_m))$. Due to Property B.2 and using the definition of c from Eq. (6.12) *we thus obtain*

$$c(X) = \inf\{h_X(\gamma) | \gamma \in \Gamma\} \geq \inf\{\eta_1 \cdot h_{X_1}(\gamma) + \cdots + \eta_m \cdot h_{X_m}(\gamma) | \gamma \in \Gamma\}. \tag{B.6b}$$

Observing that

$$\left\{ \sum_{k \in [m]} \eta_k \cdot h_{X_k}(\gamma) \,\middle|\, \gamma \in \Gamma \right\} \subseteq \left\{ \sum_{k \in [m]} \eta_k \cdot h_{X_k}(\gamma_k) \,\middle|\, \gamma_1, \ldots, \gamma_m \in \Gamma \right\}, \tag{B.6c}$$

we further obtain

$$c(X) \geq \inf\{\eta_1 \cdot h_{X_1}(\gamma_1) + \cdots + \eta_m \cdot h_{X_m}(\gamma_m) | \gamma_1, \ldots, \gamma_m \in \Gamma\} \tag{B.6d}$$
$$= \eta_1 \cdot \inf\{h_{X_1}(\gamma_1) | \gamma_1 \in \Gamma\} + \cdots + \eta_m \cdot \inf\{h_{X_m}(\gamma_m) | \gamma_m \in \Gamma\}, \tag{B.6e}$$

and thus $c(X) \geq \eta_1 \cdot c(X_1) + \cdots + \eta_m \cdot c(X_m)$. □

B.3 Proofs of Section 6.2

Below, we give the proofs for Property 6.1 and 6.2 as well as Criterion 6.1 and 6.2.

Proof of Property 6.1. According to Eq. (6.16), $\eta \in P(\mathbb{P}_k(X))$ is fulfilled by any feasible solution $\eta_1, \ldots, \eta_m$ of $\textsc{MaxSetPacking}_{\text{LP}}(\mathbb{P}_k(X))$. Since $\bigcup \mathbb{P}_k(X) = X$, Eq. (6.17) then follows directly from Property B.2 and Eq. (6.15). □

Proof of Property 6.2. There is a family $\mathscr{X} \subset \mathbb{P}(U)$ so that $\bigcup \mathscr{X} = U$ and

$$\sum_{X' \in \mathscr{X}} \tilde{c}(X') = \textsc{MinSetCover}(\mathbb{P}(U)). \tag{B.7a}$$

The condition in Eq. (6.19a) means that $\mathscr{X}$ is unique and $\exists Y \in \mathscr{X} : X \subseteq Y$. Then,

$$\textsc{MinSetCover}(\mathbb{P}(U)) = \sum_{X' \in \mathscr{X}} \tilde{c}(X') = \tilde{c}(Y) + \sum_{X' \in \mathscr{X} \setminus Y} \tilde{c}(X') \tag{B.7b}$$
$$\geq \tilde{c}(Y) + \sum_{u \in U \setminus Y} c(\{u\}) \tag{B.7c}$$
$$\geq \tilde{c}(X) + \sum_{u \in Y \setminus X} c(\{u\}) + \sum_{u \in U \setminus Y} c(\{u\}) \tag{B.7d}$$
$$\geq \tilde{c}(X) + \sum_{u \in U \setminus X} c(\{u\}) \tag{B.7e}$$

due to the superadditivity property stated in Eq. (6.14). □

Proof of Criterion 6.1. Property 6.2 states that Eq. (6.19a) ⇒ Eq. (6.19b), thus ¬ Eq. (6.19b) ⇒ ¬ Eq. (6.19a). Due to Eq. (6.20), ¬ Eq. (6.19b) is equivalent to Eq. (6.21a) and ¬ Eq. (6.19a) is equivalent to Eq. (6.21b) since $\textsc{MinSetCover}(\mathbb{P}(U)) \leq \textsc{MinSetCover}(\mathbb{P}(U) \setminus \{Y\})$. □

Proof of Criterion 6.2. Let $\tilde{c}(U) \leq 2\beta + \sum_{u \in U} c(\{u\})$ and assume that $\tilde{c}(U) > \textsc{MinSetCover}(\mathbb{P}(U))$. Then, there is a family $\mathscr{X} \subset \mathbb{P}(U)$ with cardinality $\#\mathscr{X} \geq 2$

so that $\bigcup \mathscr{X} = U$,

$$\sum_{X \in \mathscr{X}} \tilde{c}\,(X) = \textsc{MinSetCover}\,(\mathbb{P}\,(U)) < \tilde{c}\,(U) \leq 2\beta + \sum_{u \in U} c\,(\{u\}) \tag{B.8a}$$

and due to $\sum_{X \in \mathscr{X}} \tilde{c}\,(X) \geq \beta \cdot \#\mathscr{X} + \sum_{X \in \mathscr{X}} c\,(X) \geq \beta \cdot \#\mathscr{X} + \sum_{u \in U} c\,(\{u\})$,

$$\beta \cdot \#\mathscr{X} + \sum_{u \in U} c\,(\{u\}) \leq \sum_{X \in \mathscr{X}} \tilde{c}\,(X) < 2\beta + \sum_{u \in U} c\,(\{u\})\,, \tag{B.8b}$$

which is a contradiction since $\#\mathscr{X} \geq 2$, i.e. the assumption $\tilde{c}\,(U) > \textsc{MinSetCover}\,(\mathbb{P}\,(U))$ is wrong. Thus, Criterion 6.2 is proved. □

C Algorithmic and implementation details

C.1 Example run-through of Algorithm 6.1

Figure C.1 illustrates an example run-through of Algorithm 6.1 (Section 6.2.5) using the example image region in Figure C.1a. The corresponding universe $U = \{u_1, \ldots, u_5\}$ of five atomic image regions, where each region only contains image points from a single object, is shown in Figure C.1b. The corresponding adjacency graph $\mathcal{G} = (U, \mathcal{E})$ serves as the input for Algorithm 6.1. In the first iteration, the algorithm computes the min-weight set-cover $\textsc{MinSetCover}\,(\mathscr{U}_1)$ in line 4, which demands the computation of the energy $\tilde{c}\,(\{u\})$ of the singleton sets $\{u\} \in \mathbb{P}_1\,(U)$. Then, the lower bounds $\tilde{c}_{\min}$ and the upper bounds $\tilde{c}_{\max}$ of the extended set energy are computed for each connected set $X \subset U$ of cardinality $\#X = 2$. There are 8 such sets in total and each corresponds to a pair of blue/orange crosses in Figure C.1c in iteration 1. The condition $\tilde{c}_{\min} \leq \tilde{c}_{\max}$ holds for all these sets (all orange crosses are above their corresponding blue crosses). Thus, the energies $\tilde{c}\,(X)$ are then computed for all 8 sets of cardinality $\#X = 2$. In iteration 2, the algorithm computes $\textsc{MinSetCover}\,(\mathscr{U}_2)$ in line 4 and then processes the connected sets $X \subset U$ of cardinality $\#X = 3$ (there are 9 such sets in total). The results of the previous iteration are exploited to obtain bounds $\tilde{c}_{\min}$ and $\tilde{c}_{\max}$ that are *tighter* than before (the orange and blue crosses in Figure C.1c generally are closer to each other). Only 4 of the 9 sets pass the condition $\tilde{c}_{\min} \leq \tilde{c}_{\max}$ in iteration 2 ($\{u_1, u_3, u_4\}$, $\{u_1, u_3, u_5\}$, $\{u_1, u_4, u_5\}$, $\{u_3, u_4, u_5\}$). For example, the set $X = \{u_1, u_2, u_4\}$ yields $\tilde{c}_{\min} = 4220.7$ and $\tilde{c}_{\max} = 3627.7$ and is thus excluded. Moreover, *neither of its supersets* requires to be considered in subsequent iterations. This is reasonable, since the regions u_1 and u_2 truthfully correspond to different objects, and all supersets of $\{u_1, u_2, u_4\}$ would thus lead to falsely merged objects. As another example, the set $X = \{u_2, u_4, u_5\}$ yields $\tilde{c}_{\min} = 3563.6$ and $\tilde{c}_{\max} = 3681.1$ and thus passes the condition $\tilde{c}_{\min} \leq \tilde{c}_{\max}$ with a tiny margin.

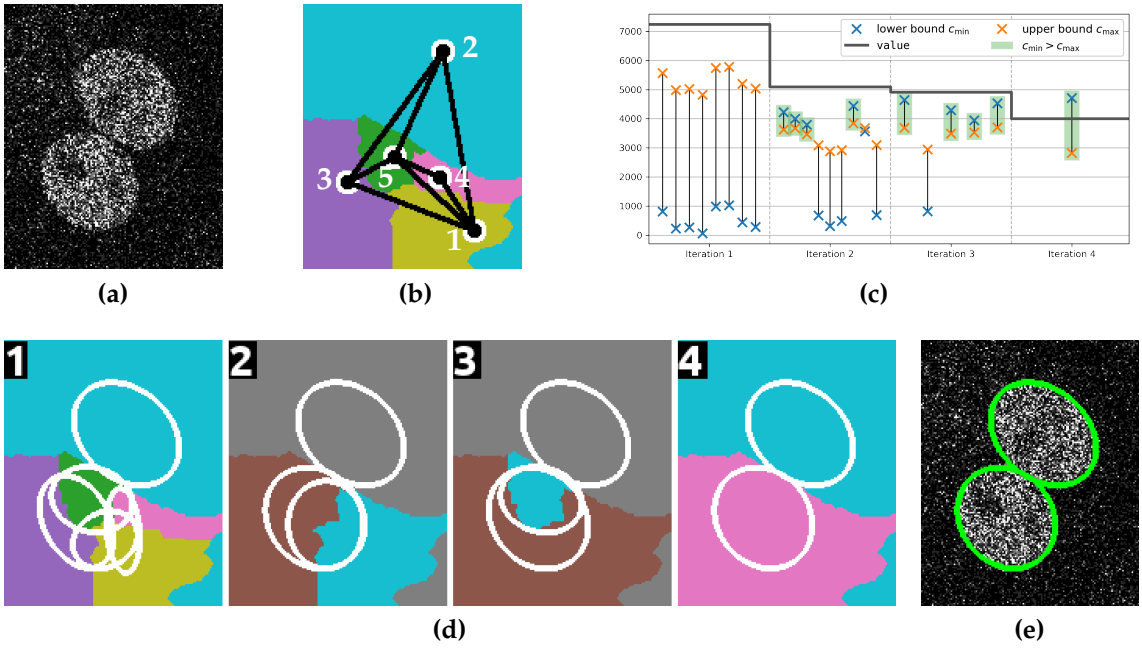


Figure C.1. Example of determining $\textsc{MinSetCover}(\mathbb{P}(U))$ by Algorithm 6.1. (a) Original image section (GOWT1 dataset 1, contrast-enhanced). (b) Computed universe U and corresponding adjacency graph. (c) Lower and upper bounds as well as the value of the objective function (the variable `value`) of Algorithm 6.1 for iterations 1 to 4. (d) Optimal set $\mathscr{X}$ from which the variable `value` is computed (line 4) for iterations 1 to 4 and corresponding intermediate segmentation results (white contours). (e) Segmentation result (green contours).

However, the computation of its extended energy yields $\tilde{c}(X) = 3917.4$, which violates the condition $\tilde{c}(X) > \tilde{c}_{\max}$, which is why this set also is excluded. In iteration 3, connected sets $X \subset U$ of cardinality $\#X = 4$ are processed. Only the set $\{u_1, u_3, u_4, u_5\}$ passes the condition $\tilde{c}_{\min} \leq \tilde{c}_{\max}$ (there is only a single blue cross below its corresponding orange crosses in Figure C.1c in iteration 3). No set passes this condition in iteration 4, which leads to `#iter1` $= 0$, and the algorithm terminates. Figure C.1d shows the optimal $\mathscr{X}$ from which $\textsc{MinSetCover}(\mathscr{U}_k)$ is computed for the iterations $k = 1, \ldots, 4$. Figure C.1e shows the final segmentation result, which corresponds to the family $\mathscr{X} = \{\{u_1, u_3, u_4, u_5\}, \{u_2\}\}$.

C.2 Coarse-to-fine region analysis

The coarse-to-fine region analysis scheme (in Section 6.3.2) determines the universe U for regions of possibly clustered objects, subject to the two constraints that *(i)* no region is smaller than a circle of radius `min_region_radius` and *(ii)* each split improves the normalized energy r in Eq. (6.23) at least by the factor `min_norm_energy_improvement`. Regions ω, for which $r(\omega) \leq$ `max_norm_energy1` are not split further. We used `max_norm_energy1` $= 0.05$, `min_norm_energy_im-`

provement = 0.1, and min_region_radius= $0.33\,\sigma \cdot \sqrt{2}$, where σ is the estimated scale. The algorithm is given as pseudo-code in Algorithm C.1. $\mathrm{dist}_S(x) = \min_{x' \in S} \|x - x'\|_2$ is the distance of the image point x to the set S and $\mathrm{split}_\omega(x_0, x_1)$ is the watershed transform of Y_ω seeded by x_0 and x_1. Each region is associated with a seed point. The seed point is used as x_0 for splitting and is also employed as seed point for the newly formed region u_2 (line 12). As seeds we use intensity peaks, and each seed is considered at most once. Thus, the number of iterations is upper-bound by the number of local intensity peaks. Regions are not split further if the normalized energy is below or equal max_norm_energy1, if splitting would violate the size constraint, or if no seeds are left.

An example run-through of Algorithm C.1 is given in Figure C.3 using the image region shown in Figure C.2a. Subfigure 0 of Figure C.3 shows the state of the algorithm right after the initialization (before the first iteration, seed point 1 corresponds to x in line 4). The normalized energy of the set ω is 0.416, which is why the tuple (x, ω) is enqueued into Q (line 5). Afterwards, the loop of the algorithm starts. Iteration 1 considers the entire set ω for splitting (subfigure 1) and chooses seed point 2 as the point in X, which is farthest away from x (corresponds to x_1 in line 8). Splitting the whole region using the two seed points yields the two depicted regions (which correspond to u_1 and u_2 in line 12). However, the improvement of the normalized energy is insufficient (line 15), which is why this split is discarded (seed point 1 and the whole region ω are re-enqueued). Iteration 2 (subfigure 2) considers a different seed point for splitting the image region ω, which again is farthest away from all seed points considered so far (seed point 3). Using seed point 1 and seed point 3 yields sufficiently improved normalized energies for both newly formed image regions u_1 and u_2. Since both normalized energies $r(u_1) = 0.122$ and $r(u_2) = 0.345$ still exceed max_norm_energy1 = 0.05, both are enqueued for further splitting (lines 17 and 19). One of these two regions is considered in iteration 3 (subfigure 3). Here, seed point 4 is chosen for splitting. The split yields $r(u_2) = 0.024$ for the left image region and $r(u_1) = 0.054$ for the right image region. Since the normalized energy of the left image region is sufficiently low, u_2 is added to the universe U (line 20), but (x_1, u_1) is enqueued for further splitting (line 17). Iteration 4 adds the image region associated with seed point 5 to the universe U (subfigure 4). In the subsequent iterations, further seed points are processed as candidates for splitting, but neither split yields a sufficient improvement of the normalized energy (subfigures 5–8). In iteration 9 (subfigure 9), the algorithm is supposed to split the region associated with seed point 4. However, no further unused seed point exists in that region, therefore this region is added to the universe U (line 10), although its normalized energy is slightly higher than max_norm_energy1 = 0.05. Iterations 10 and 11 finally add three more regions to the universe U. The finally obtained universe U as well as the corresponding adjacency graph $\mathcal{G} = (U, \mathcal{E})$ are shown in Figure C.2c. Note that, for simplicity, we have skipped iterations corresponding to violations of the

Algorithm C.1: Coarse-to-fine analysis of a region of possibly clustered objects.

input: Image region $\omega \subseteq \Omega$, `max_norm_energy1`, `min_norm_energy_improvement`, `min_region_radius`

$X \leftarrow \left\{x \in \omega \cap \Omega_{\text{fg}} : Y_\omega|_{\omega=\{x\}} \text{ is maximal within } 3 \times 3 \text{ neighborhood}\right\}$; *// set of potential seed points*

$U \leftarrow \emptyset$; $S \leftarrow \emptyset$; $Q \leftarrow$ empty queue; *// initialization*

if $r(\omega) >$ `max_norm_energy1` **then**

  $x \leftarrow \arg\max_{x \in X} Y_\omega|_{\omega=\{x\}}$; *// pick the seed point x with the highest intensity*

  Insert (x, ω) into Q and x into S; *// associate the region ω with x and mark x as chosen*

while $\#Q > 0$ **do**

  $x_0, u_0 \leftarrow \text{pop}(Q)$; *// pick a region and its associated seed point*

  $x_1 \leftarrow \arg\max_{x \in X \cap u_0} \text{dist}_S(x)$; *// pick seed point in u_0 farthest from previously chosen seeds*

  Insert x_1 into S; *// mark seed point*

  x_1 as chosen **if** $\#u_0 < 2\pi \cdot$ `min_region_radius`^{2} or $\text{dist}_S(x_1) = 0$ **then** Insert u_0 into U; *// cannot split u_0 further*

  else

    $u_2, u_1 \leftarrow \text{split}_{u_0}(x_0, x_1)$; *// split u_0 using seeds x_0 and x_1*

    if $\#u_2 < \pi \cdot$ `min_region_radius`^{2} **then** Insert (x_1, u_0) into Q; *// re-associate u_0 with x_1*

    else if $\#u_1 < \pi \cdot$ `min_region_radius`^{2} **then** Insert (x_0, u_0) into Q;

    else if $1 - \max\{r(u_1), r(u_2)\} / r(u_0) <$ `min_norm_energy_improvement` **then** Insert (x_0, u_0) into Q;

    else

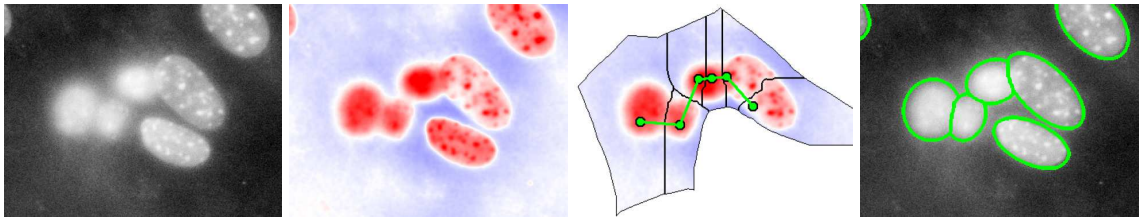
      if $r(u_1) >$ `max_norm_energy1` **then** Insert (x_1, u_1) into Q; *// associate u_1 with x_1*

      else Insert u_1 into U;

      if $r(u_2) >$ `max_norm_energy1` **then** Insert (x_0, u_2) into Q; *// associate u_2 with x_0*

      else Insert u_2 into U;

return the universe U;



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

Figure C.2. Example region of possibly clustered objects. (a) Original image section (NIH3T3 cells, contrast-enhanced). (b) Image intensities with offset τ_x (blue and red correspond to negative and positive intensities, respectively). (c) Elements of the universe U (delineated by black lines) and corresponding adjacency graph (green). (d) Segmentation result of our method.

size constraint. For example, such iterations occur between subfigures 4 and 5, which leads to a shift of seed point 4 (due to line 13, cf. subfigure 9).

C.3 Efficient implementations of the extended set energy functions and Algorithm 6.1

The computation of the extended set energy function $\tilde{c}(X) = \inf_{\theta,\xi} \psi_{\tilde{\omega}(X)}(\theta, \xi) + \beta$ requires solving the convex problem in Eq. (6.7). As described in Section 6.2.2, a globally optimal solution is obtained for any initialization due to Property B.1. However, for faster convergence, we solve the convex problem in Eq. (6.7) in two steps. First, the optimal *elliptical* model parameters θ_{ell} of the convex energy $\psi_{\tilde{\omega}(X)}(\theta, \mathbb{0})$ with respect to θ are computed. Second, $\theta = \theta_{\text{ell}}$ and $\xi = \mathbb{0}$ are used as the initialization for the minimization of the convex energy $\psi_{\tilde{\omega}(X)}(\theta, \xi)$ for *deformable* shape models. This two-step scheme is faster than the direct solution of Eq. (6.7) for two reasons: First, θ is a vector of dimension 6, whereas ξ generally is of the much higher dimension $\#\Omega$. Thus, minimization with respect to θ is much faster than minimization with respect to θ and ξ. The second reason is that, if the object within the image region represented by the set X is almost elliptical, then the optimal vector ξ is also close to $\mathbb{0}$, i.e.

$$\inf_{\theta,\xi} \psi_{\tilde{\omega}(X)}(\theta, \xi) \approx \psi_{\tilde{\omega}(X)}(\theta_{\text{ell}}, \mathbb{0}) = \inf_{\theta} \psi_{\tilde{\omega}(X)}(\theta, \mathbb{0}) \,. \tag{C.1}$$

This means that the initialization of the second step is close to the optimal solution. In the ideal case when the object is *exactly* elliptical, the initialization of the second step *is* the optimal solution, i.e. $\inf_{\theta,\xi} \psi_{\tilde{\omega}(X)}(\theta, \xi) = \psi_{\tilde{\omega}(X)}(\theta_{\text{ell}}, \mathbb{0})$.

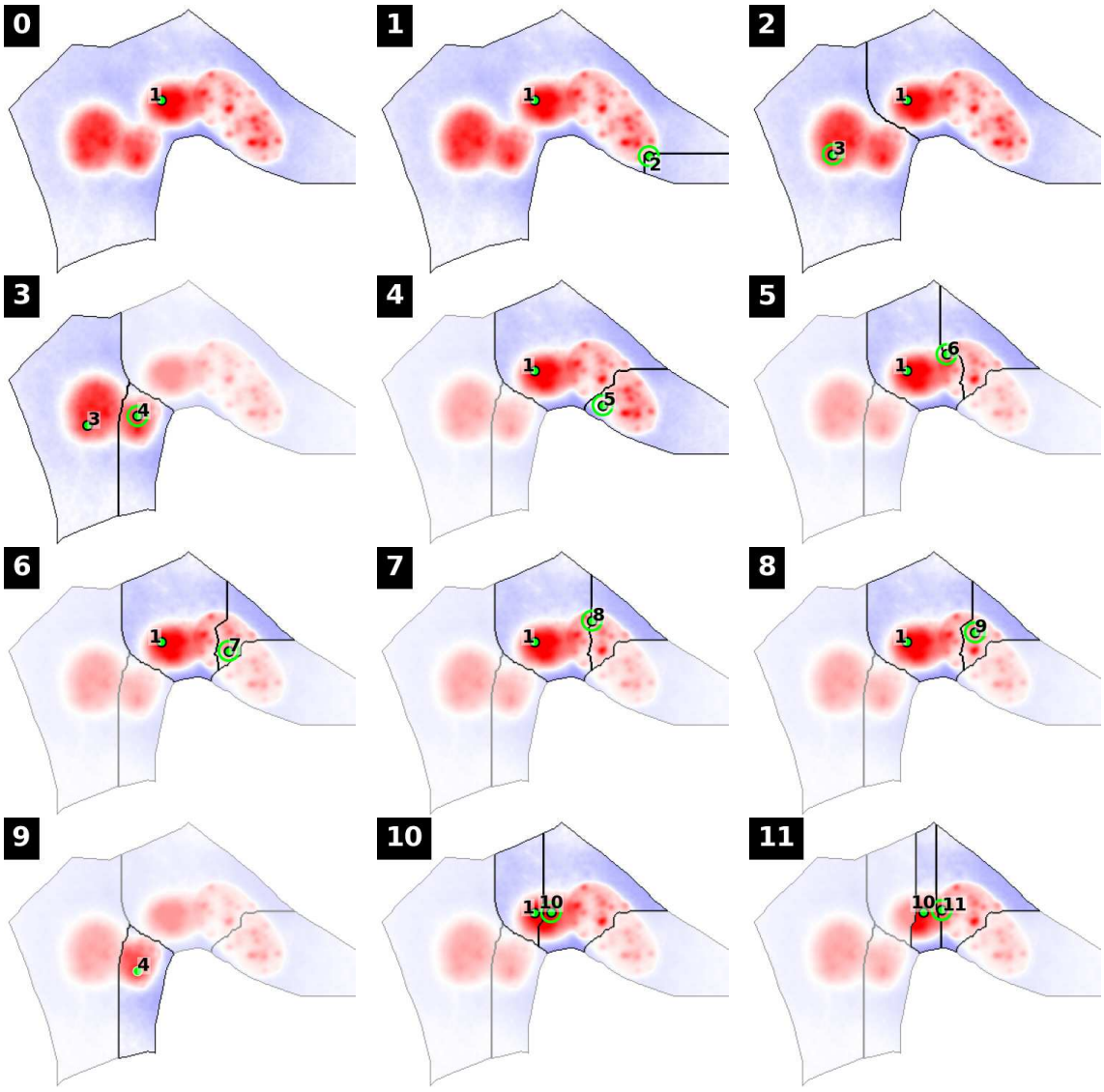


Figure C.3. Intermediate steps of the coarse-to-fine region analysis scheme, which determines the universe U for a region of possibly clustered objects (Algorithm C.1). Subfigure k shows the state after the k-th iteration ($k = 0$ corresponds to the state before the first iteration). The green dots correspond to the seed points associated with the respective image regions (the circled dot marks the newly chosen seed point).

To further accelerate convex programming, the size of image background included in $\tilde{\omega}(X)$ is reduced. Analogously to Eq. (6.11) we use

$$\begin{gathered}\inf_{\theta,\xi} \psi_{\tilde{\omega}(X)}(\theta,\xi) \approx \inf_{\theta,\xi} \psi_{\tilde{\omega}'(X)}(\theta,\xi), \\ \text{where } \tilde{\omega}'(X) = \left\{ x \in \tilde{\omega}(X) \middle| \operatorname{dist}_{\Omega_{\mathrm{fg}}}(x) \leq \sigma_G \right\}\end{gathered} \tag{C.2}$$

and $\operatorname{dist}_{\Omega_{\mathrm{fg}}}(x) = \min_{x' \in \Omega_{\mathrm{fg}}} \|x - x'\|_2$ is the Euclidean distance of the image point x to the set C.

For efficient implementation of Algorithm 6.1, we include the set U into the family $\mathscr{U}$ of Algorithm 6.1 from the start, since the extended set energy $\tilde{c}(U)$ is anyway computed in advance for evaluation of Criterion 6.2 (cf. Section 6.3.3), and the iteration corresponding to `iter1` $= \{U\}$ is thus skipped.

C.4 Details on post-processing (Section 6.3.4)

Each element of the family $\mathscr{X}$ is an image region $X \subseteq U$, which represents a θ, ξ-parameterized deformable shape model S. The optimal parameters θ and ξ are determined by convex programming (Section 6.3.3). The corresponding segmentation mask $M = \mathcal{I}_S^+(\theta, \xi) \cap \tilde{\omega}(X)$ is obtained from the zero-superlevel set $\mathcal{I}_S^+(\theta, \xi)$ of the model S. Each segmentation mask $M \subset \Omega$ is refined individually. First, morphological holes in M are filled. Then, image points $x \in \Omega$ within a maximum distance of `mask_max_distance` to the boundary of the segmentation mask M are added to the segmentation mask if

$$-\texttt{mask\_stdamp} \cdot \operatorname*{std}_{x' \in M} g'_{x'} \leq g'_x - \operatorname*{mean}_{x' \in M} g'_{x'} \leq +\texttt{mask\_stdamp} \cdot \operatorname*{std}_{x' \in M} g'_{x'} \tag{C.3}$$

and removed otherwise. Here, $\operatorname{mean}_{x' \in M} g'_{x'}$ and $\operatorname{std}_{x' \in M} g'_{x'}$ denote the arithmetic average and the standard deviation of $g'_{x'}$ for $x' \in M$, where g'_x and $g'_{x'}$ are the intensity values of the Gaussian-filtered image at image points x and x', respectively (using a Gaussian filter with standard deviation 3). We used the default values `mask_max_distance` $= 1$ and `mask_stdamp` $= 2$ in our experiments.

In addition, spurious objects are discarded based on a set of criteria. An object $X \in \mathscr{X}$ and its corresponding segmentation mask M are discarded if at least one of the following rejection criteria is fulfilled:

1. The object is fully contained in the image and the eccentricity of its boundary is larger than 0.99.

2. The normalized energy $c(X)/\#\tilde{\omega}'(X)$ is larger than `min_norm_energy2`.

3. The ratio of the mean image intensity inside the mask and the mean image intensity within its neighborhood is smaller than `min_contrast` (i.e. the contrast is too low).

4. The radius $\sqrt{\#M/\pi}$ of a circle of the same size $\#M$ as the object mask M is smaller than `min_object_radius` (i.e. the object is too small).

D Hyperparameters used in the experiments (Section 6.4)

Table D.1. Hyperparameters used in the experiments ("—" for SuperDSM* indicates that the default values were used, see the column SuperDSM). The value $\sigma = 42.43$ for SuperDSM* is the mean scale σ estimated over *all* images of the dataset using SuperDSM. The parameter `max_pa_ratio` denotes the maximum P/A ratio described in Section 6.3.1.

Processing step	**Parameter**	**SuperDSM**	**SuperDSM***						
		Defaults	NIH3T3	U2OS	GOWT1 dataset 1	GOWT1 dataset 2	Fibroblast	HeLa	H&E histopathology data
Pre-processing (Section 6.3.1)	σ	Automatic	40	—	42.43	—	80	70	5
Coarse-to-fine region analysis (Section 6.3.2)	`min_region_radius`	Automatic	—	—	—	—	—	—	10
	`min_norm_energy_improvement`	0.1	—	—	$-\infty$	—	—	0	$-\infty$
	`max_norm_energy1`	0.05	—	—	—	—	—	—	—
	`max_pa_ratio`	0.2	—	—	—	—	—	—	0.5
Global energy minimization (Section 6.3.3)	α	Automatic	0.6	—	—	0.1	—	1.5	0.4
	β	Automatic	1200	—	—	—	—	3500	75
	$\alpha_{\texttt{factor}}$	$5 \cdot 10^{-4}$	—	$7.5 \cdot 10^{-5}$	—	—	—	—	—
	$\beta_{\texttt{factor}}$	0.33	—	0.15	—	—	—	—	—
Post-processing (Section 6.3.4)	`mask_max_distance`	1	—	2	2	—	—	—	5
	`mask_stdamp`	2	3	3	1	—	—	—	—
	`max_norm_energy2`	0.2	—	—	0.5	0.5	—	—	—
	`min_contrast`	1.35	1.25	—	1.45	1.45	1.8	—	—
	`min_object_radius`	0	—	—	—	—	—	40	—

Bibliography

[1] M. Zwerger, C. Y. Ho, and J. Lammerding, "Nuclear mechanics in disease," *Annu. Rev. Biomed. Eng.*, vol. 13, pp. 397–428, 2011.

[2] H. J. Worman, L. G. Fong, A. Muchir, S. G. Young *et al.*, "Laminopathies and the long strange trip from basic cell biology to therapy," *J. Clin. Investig.*, vol. 119, no. 7, pp. 1825–1836, 2009.

[3] C. Denais and J. Lammerding, "Nuclear mechanics in cancer," in *Cancer Biology and the Nuclear Envelope: Recent Advances May Elucidate Past Paradoxes*, E. C. Schirmer and J. I. de las Heras, Eds. New York, NY: Springer, 2014, ch. 6, pp. 435–470.

[4] D. Zink, A. H. Fischer, and J. A. Nickerson, "Nuclear structure in cancer cells," *Nat. Rev. Cancer*, vol. 4, no. 9, pp. 677–687, 2004.

[5] K. Khoshmanesh, A. Z. Kouzani, S. Nahavandi, S. Baratchi, and J. R. Kanwar, "At a glance: Cellular biology for engineers," *Comput. Biol. Chem.*, vol. 32, no. 5, pp. 315–331, 2008.

[6] M. Mulisch and U. Welsch, Eds., *Romeis - Mikroskopische Technik*. Springer-Verlag, 2015.

[7] C. Aguerrebere, J. Delon, Y. Gousseau, and P. Musé, "Study of the digital camera acquisition process and statistical modeling of the sensor raw data," *HAL preprint hal-00733538v4*, 2013.

[8] Y. Zhang, Y. Zhu, E. Nichols, Q. Wang, S. Zhang, C. Smith, and S. Howard, "A Poisson-Gaussian denoising dataset with real fluorescence microscopy images," in *Proc. CVPR 2019*. IEEE, 2019, pp. 11 710–11 718.

[9] P. A. Morris, R. S. Aspden, J. E. Bell, R. W. Boyd, and M. J. Padgett, "Imaging with a small number of photons," *Nat. Commun.*, vol. 6, no. 1, pp. 1–6, 2015.

[10] F. Pontvianne, M.-C. Carpentier, N. Durut, V. Pavlištová, K. Jaške, Š. Schořová, H. Parrinello, M. Rohmer, C. S. Pikaard, M. Fojtová *et al.*, "Identification of nucleolus-associated chromatin domains reveals a role for the nucleolus in 3d organization of the a. thaliana genome," *Cell Rep.*, vol. 16, no. 6, pp. 1574–1587, 2016.

[11] L. Kostrykin, C. Schnörr, and K. Rohr, "Segmentation of cell nuclei using intensity-based model fitting and sequential convex programming," in *Proc. ISBI 2018*. IEEE, 2018, pp. 654–657.

[12] L. Kostrykin, C. Schnörr, and K. Rohr, "Globally optimal segmentation of cell nuclei in fluorescence microscopy images using shape and intensity information," *Med. Image Anal.*, vol. 58, 101536, 2019.

[13] L. Kostrykin and K. Rohr, "Superadditivity and convex optimization for globally optimal cell segmentation using deformable shape models," *IEEE Trans. Pattern Anal. Mach. Intell.*, *submitted*, *minor revision*, 2022.

[14] R. T. Rockafellar, *Convex analysis*. Princeton University Press, 1970.

[15] S. Boyd and L. Vandenberghe, *Convex optimization*. Cambridge University Press, 2004.

[16] D. Bertsekas, *Convex optimization theory*. Athena Scientific, 2009.

[17] G. Ausiello, P. Crescenzi, G. Gambosi, V. Kann, A. Marchetti-Spaccamela, and M. Protasi, *Complexity and approximation: Combinatorial optimization problems and their approximability properties*. Springer Science & Business Media, 1999.

[18] A. Cobham, "The intrinsic computational difficulty of functions," in *Proc. LMPS 1965*. North-Holland Publishing, 1965, pp. 24–30.

[19] J. Edmonds, "Paths, trees, and flowers," *Can. J. Math.*, vol. 17, pp. 449–467, 1965.

[20] S. Basu, R. Pollack, and M.-F. Roy, *Algorithms in Real Algebraic Geometry*. Springer, 2006.

[21] J. Canny, "Some algebraic and geometric computations in PSPACE," in *Proc. STOC 1988*, 1988, pp. 460–467.

[22] K. P. Murphy, *Machine Learning: A Probabilistic Perspective*. MIT press, 2012.

[23] P. L. Bartlett, M. I. Jordan, and J. D. McAuliffe, "Convexity, classification, and risk bounds," *J. Am. Stat. Assoc.*, vol. 101, no. 473, pp. 138–156, 2006.

[24] A. El-Bakry, R. A. Tapia, T. Tsuchiya, and Y. Zhang, "On the formulation and theory of the newton interior-point method for nonlinear programming," *J. Optim. Theory Appl.*, vol. 89, no. 3, pp. 507–541, 1996.

[25] S. Jiang, Z. Song, O. Weinstein, and H. Zhang, "Faster dynamic matrix inverse for faster LPs," *arXiv preprint arXiv:2004.07470*, 2020.

[26] V. Vazirani, *Approximation algorithms*. Berlin New York: Springer, 2001.

[27] D. S. Johnson, "Approximation algorithms for combinatorial problems," *J. Computer Syst. Sciences*, vol. 9, no. 3, pp. 256–278, 1974.

[28] Y. LeCun, S. Chopra, R. Hadsell, M. Ranzato, and F. Huang, "Energy-based models," in *Predicting structured data*, G. Bakir, T. Hofmann, B. Schölkopf, A. Smola, B. Taskar, and S. Vishwanathan, Eds. Cambridge, MA, USA: MIT Press, Sep. 2007, ch. 10, pp. 191–246.

[29] K. Wu, D. Gauthier, and M. D. Levine, "Live cell image segmentation," *IEEE Trans. Biomed. Eng.*, vol. 42, no. 1, pp. 1–12, 1995.

[30] S. Singh, F. Janoos, T. Pécot, E. Caserta, K. Huang, J. Rittscher, G. Leone, and R. Machiraju, "Non-parametric population analysis of cellular phenotypes," in *Proc. MICCAI 2011*. Springer, 2011, pp. 343–351.

[31] R. Singh, M. Sharma, and M. Bhattacharya, "Segmentation of heavily clustered cell nuclei in histopathological images," in *Proc. ECCOMAS 2017*. Springer, 2017, pp. 244–254.

[32] Y. He, Y. Meng, H. Gong, S. Chen, B. Zhang, W. Ding, Q. Luo, and A. Li, "An automated three-dimensional detection and segmentation method for touching cells by integrating concave points clustering and random walker algorithm," *PloS One*, vol. 9, no. 8, e104437, 2014.

[33] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE Trans. Syst., Man, Cybern.*, vol. 9, no. 1, pp. 62–66, 1979.

[34] M. Ester, H.-P. Kriegel, J. Sander, X. Xu *et al.*, "A density-based algorithm for discovering clusters in large spatial databases with noise," in *Proc. KDD 1996*, vol. 96, no. 34, 1996, pp. 226–231.

[35] M. Daszykowski, B. Walczak, and D. Massart, "Looking for natural patterns in data: Part 1. density-based approach," *Chemometr. Intell. Lab. Syst.*, vol. 56, no. 2, pp. 83–92, 2001.

[36] C. Wählby, J. Lindblad, M. Vondrus, E. Bengtsson, and L. Björkesten, "Algorithms for cytoplasm segmentation of fluorescence labelled cells," *Anal. Cell. Pathol.*, vol. 24, no. 2-3, pp. 101–111, 2002.

[37] J. Cheng, J. C. Rajapakse *et al.*, "Segmentation of clustered nuclei with shape markers and marking function," *IEEE Trans. Biomed. Eng.*, vol. 56, no. 3, pp. 741–748, 2008.

[38] M. E. Plissiti, C. Nikou, and A. Charchanti, "Automated detection of cell nuclei in pap smear images using morphological reconstruction and clustering," *IEEE Trans. Inf. Technol. Biomed.*, vol. 15, no. 2, pp. 233–241, 2011.

[39] M. Kass, A. Witkin, and D. Terzopoulos, "Snakes: Active contour models," *Int. J. Comput. Vis.*, vol. 1, no. 4, pp. 321–331, 1988.

[40] C. Zimmer and J.-C. Olivo-Marin, "Coupled parametric active contours," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 11, pp. 1838–1842, 2005.

[41] P. Thevenaz and M. Unser, "Snakuscules," *IEEE Trans. Image Process.*, vol. 17, no. 4, pp. 585–593, 2008.

[42] P. Thevenaz, R. Delgado-Gonzalo, and M. Unser, "The Ovuscule," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 2, pp. 382–393, 2011.

[43] R. Delgado-Gonzalo, P. Thévenaz, C. S. Seelamantula, and M. Unser, "Snakes with an ellipse-reproducing property," *IEEE Trans. Image Process.*, vol. 21, no. 3, pp. 1258–1271, 2012.

[44] R. Delgado-Gonzalo, N. Chenouard, and M. Unser, "Spline-based deforming ellipsoids for interactive 3d bioimage segmentation," *IEEE Trans. Image Process.*, vol. 22, no. 10, pp. 3926–3940, 2013.

[45] S. Osher and J. A. Sethian, "Fronts propagating with curvature-dependent speed: Algorithms based on hamilton-jacobi formulations," *J. Comput. Phys.*, vol. 79, no. 1, pp. 12–49, 1988.

[46] D. B. Mumford and J. Shah, "Optimal approximations by piecewise smooth functions and associated variational problems," *Commun. Pure Appl. Math.*, vol. 42, no. 5, pp. 577–685, 1989.

[47] T. F. Chan and L. A. Vese, "Active contours without edges," *IEEE Trans. Image Process.*, vol. 10, no. 2, pp. 266–277, 2001.

[48] C. de Solorzano, R. Malladi, S. Lelievre, and S. Lockett, "Segmentation of nuclei and cells using membrane related protein markers," *J. Microsc.*, vol. 201, no. 3, pp. 404–415, 2001.

[49] A. Dufour, V. Shinin, S. Tajbakhsh, N. Guillén-Aghion, J.-C. Olivo-Marin, and C. Zimmer, "Segmenting and tracking fluorescent cells in dynamic 3-D microscopy with coupled active surfaces," *IEEE Trans. Image Process.*, vol. 14, no. 9, pp. 1396–1410, 2005.

[50] C. Li, C.-Y. Kao, J. C. Gore, and Z. Ding, "Minimization of region-scalable fitting energy for image segmentation," *IEEE Trans. Image Process.*, vol. 17, no. 10, pp. 1940–1949, Oct. 2008.

[51] J.-P. Bergeest and K. Rohr, "Efficient globally optimal segmentation of cells in fluorescence microscopy images using level sets and convex energy functionals," *Med. Image Anal.*, vol. 16, no. 7, pp. 1436–1444, 2012.

[52] A. Gharipour and A. W.-C. Liew, "Segmentation of cell nuclei in fluorescence microscopy images: An integrated framework using level set segmentation and touching-cell splitting," *Pattern Recognit.*, vol. 58, pp. 1–11, 2016.

[53] T. F. Chan, S. Esedoglu, and M. Nikolova, "Algorithms for finding global minimizers of image segmentation and denoising models," *SIAM J. Appl. Math.*, vol. 66, no. 5, pp. 1632–1648, 2006.

[54] T. Goldstein, X. Bresson, and S. Osher, "Geometric applications of the split Bregman method: Segmentation and surface reconstruction," *J. Sci. Comput.*, vol. 45, no. 1, pp. 272–293, 2010.

[55] M. Rousson and R. Deriche, "A variational framework for active and adaptative segmentation of vector valued images," in *Proc. MOTION 2002*. IEEE, 2002, pp. 56–61.

[56] S. Ali and A. Madabhushi, "An integrated region-, boundary-, shape-based active contour for multiple object overlap resolution in histological imagery," *IEEE Trans. Med. Imag.*, vol. 31, no. 7, pp. 1448–1460, 2012.

[57] J. Kong, P. Zhang, Y. Liang, G. Teodoro, D. Brat, and F. Wang, "Robust cell segmentation for histological images of Glioblastoma," in *Proc. ISBI 2016*. IEEE, 2016, pp. 1041–1045.

[58] M. Nosrati and G. Hamarneh, "Segmentation of overlapping cervical cells: a variational method with star-shape prior," in *Proc. ISBI 2015*. IEEE, 2015, pp. 186–189.

[59] F. Xing and L. Yang, "Fast cell segmentation using scalable sparse manifold learning and affine transform-approximated active contour," in *Proc. MICCAI 2015*. Springer, 2015, pp. 332–339.

[60] P. Zhang, F. Wang, G. Teodoro, Y. Liang, D. Brat, and J. Kong, "Automated level set segmentation of histopathologic cells with sparse shape prior support and dynamic occlusion constraint," in *Proc. ISBI 2017*. IEEE, 2017, pp. 718–722.

[61] Z. Lu, G. Carneiro, and A. P. Bradley, "An improved joint optimization of multiple level set functions for the segmentation of overlapping cervical cells," *IEEE Trans. Image Process.*, vol. 24, no. 4, pp. 1261–1272, Apr. 2015.

[62] G. Dong and S. Acton, "Detection of rolling leukocytes by marked point processes," *J. Electron. Imag.*, vol. 16(3), 033013, 2007.

[63] E. Soubies, P. Weiss, and X. Descombes, "Graph cut based segmentation of predefined shapes: Applications to biological imaging," in *Pattern Recognit. Applic. and Methods*. Springer, 2015, pp. 153–170.

[64] X. Descombes, "Multiple objects detection in biological images using a marked point process framework," *Methods*, vol. 115, pp. 2–8, 2017.

[65] C. Panagiotakis and A. Argyros, "Cell segmentation via region-based ellipse fitting," in *Proc. ICIP 2018*. IEEE, 2018, pp. 2426–2430.

[66] C. Panagiotakis and A. Argyros, "Region-based fitting of overlapping ellipses and its application to cells segmentation," *Image Vis. Comput.*, vol. 93, 103810, 2020.

[67] P. Markowsky, S. Reith, T. Zuber, R. König, K. Rohr, and C. Schnörr, "Segmentation of cell structures using model-based set covering with iterative reweighting," in *Proc. ISBI 2017*. IEEE, 2017, pp. 392–396.

[68] C. Ducroz, J.-C. Olivo-Marin, and A. Dufour, "Spherical harmonics based extraction and annotation of cell shape in 3D time-lapse microscopy sequences," in *Proc. EMBC 2011*. IEEE, 2011, pp. 6619–6622.

[69] S. Eck, S. Wörz, K. Müller-Ott, M. Hahn, A. Biesdorf, G. Schotta, K. Rippe, and K. Rohr, "A spherical harmonics intensity model for 3D segmentation and 3D shape analysis of heterochromatin foci," *Med. Image Anal.*, vol. 32, pp. 18–31, 2016.

[70] S. N. Chiu, D. Stoyan, W. S. Kendall, and J. Mecke, *Stochastic geometry and its applications*. John Wiley & Sons, 2013.

[71] P. J. Green, "Reversible jump Markov chain Monte Carlo computation and bayesian model determination," *Biometrika*, vol. 82, no. 4, pp. 711–732, 1995.

[72] Y. Boykov and G. Funka-Lea, "Graph cuts and efficient ND image segmentation," *Internat. J. Computer Vis.*, vol. 70, no. 2, pp. 109–131, 2006.

[73] A. G. Eldin, X. Descombes, G. Charpiat, and J. Zerubia, "Multiple birth and cut algorithm for multiple object detection," *J. Multimed. Processing Technol.*, 2012.

[74] C. Bouttier and I. Gavra, "Convergence rate of a simulated annealing algorithm with noisy observations," *J. Mach. Learn. Res.*, vol. 20, no. 1, pp. 127–171, 2019.

[75] A. Biesdorf, S. Wörz, H. von Tengg-Kobligk, K. Rohr, and C. Schnörr, "3D segmentation of vessels by incremental implicit polynomial fitting and convex optimization," in *Proc. ISBI 2015*. IEEE 2015, 2015, pp. 1540–1543.

[76] Y. Al-Kofahi, W. Lassoued, W. Lee, and B. Roysam, "Improved automatic detection and segmentation of cell nuclei in histopathology images," *Trans. Biomed. Eng.*, vol. 57, no. 4, pp. 841–852, 2010.

[77] X. Lou, U. Köthe, J. Wittbrodt, and F. A. Hamprecht, "Learning to segment dense cell nuclei with shape prior," in *Proc. CVPR 2012*. IEEE, 2012, pp. 1012–1018.

[78] E. Poulain, S. Prigent, E. Soubies, and X. Descombes, "Cells detection using segmentation competition," in *Proc. ISBI 2015*. IEEE, 2015, pp. 1208–1211.

[79] S. Akram, J. Kannala, L. Eklund, and J. Heikkilä, "Joint cell segmentation and tracking using cell proposals," in *Proc. ISBI 2016*. IEEE, 2016, pp. 920–924.

[80] S. Akram, J. Kannala, L. Eklund, and J. Heikkilä, "Cell tracking via proposal generation and selection," *arXiv preprint arXiv:1705.03386*, 2017.

[81] E. Türetken, X. Wang, C. Becker, C. Haubold, and P. Fua, "Network flow integer programming to track elliptical cells in time-lapse sequences," *IEEE Trans. Med. Imag.*, vol. 36, no. 4, pp. 942–951, 2017.

[82] R. B. Potts, "Some generalized order-disorder transformations," *Math. Proc. Camb. Philos. Soc.*, vol. 48, no. 1, pp. 106–109, 1952.

[83] R. Kinderman and J. L. Snell, *Markov random fields and their applications*. Providence, RI, USA: American Mathematical Society, 1980.

[84] Y. Boykov, O. Veksler, and R. Zabih, "Fast approximate energy minimization via graph cuts," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 23, no. 11, pp. 1222–1239, 2001.

[85] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[86] O. Ronneberger, P. Fischer, and T. Brox, "U-Net: Convolutional networks for biomedical image segmentation," in *Proc. MICCAI 2015*. Springer, 2015, pp. 234–241.

[87] Y. Song, E.-L. Tan, X. Jiang, J.-Z. Cheng, D. Ni, S. Chen, B. Lei, and T. Wang, "Accurate cervical cell segmentation from overlapping clumps in pap smear images," *IEEE Trans. Med. Imag.*, vol. 36, no. 1, pp. 288–300, Jan. 2017.

[88] A. Böhm, A. Ücker, T. Jäger, O. Ronneberger, and T. Falk, "ISOO DL: Instance segmentation of overlapping biological objects using deep learning," in *Proc. ISBI 2018*. IEEE, 2018, pp. 1225–1229.

[89] M. Fan and J. Rittscher, "Global probabilistic models for enhancing segmentation with convolutional networks," in *Proc. ISBI 2018*. IEEE, 2018, pp. 1234–1238.

[90] W. Xie, J. A. Noble, and A. Zisserman, "Microscopy cell counting and detection with fully convolutional regression networks," *Comput. Methods Biomech. Biomed. Eng. Imaging Vis.*, vol. 6, no. 3, pp. 283–292, 2018.

[91] A. O. Vuola, S. U. Akram, and J. Kannala, "Mask-RCNN and U-Net ensembled for nuclei segmentation," in *Proc. ISBI 2019*, 2019, pp. 208–212.

[92] C. Payer, D. Štern, M. Feiner, H. Bischof, and M. Urschler, "Segmenting and tracking cell instances with cosine embeddings and recurrent hourglass networks," *Med. Image Anal.*, vol. 57, pp. 106–119, 2019.

[93] H. Fehri, A. Gooya, Y. Lu, E. Meijering, S. A. Johnston, and A. F. Frangi, "Bayesian polytrees with learned deep features for multi-class cell segmentation," *IEEE Trans. Image Process.*, vol. 28, no. 7, pp. 3246–3260, 2019.

[94] T. Wollmann, M. Gunkel, I. Chung, H. Erfle, K. Rippe, and K. Rohr, "GRUU-net: Integrated convolutional and gated recurrent neural network for cell segmentation," *Med. Image Anal.*, vol. 56, pp. 68–79, 2019.

[95] C. Stringer, T. Wang, M. Michaelos, and M. Pachitariu, "Cellpose: a generalist algorithm for cellular segmentation," *Nat. Methods*, vol. 18, no. 1, pp. 100–106, 2020.

[96] O. Ciga and A. L. Martel, "Learning to segment images with classification labels," *Med. Image Anal.*, vol. 68, 101912, 2021.

[97] K. He, G. Gkioxari, P. Dollár, and R. Girshick, "Mask R-CNN," in *Proc. ICCV 2017*. IEEE, 2017, pp. 2961–2969.

[98] T.-Y. Lin, P. Goyal, R. Girshick, K. He, and P. Dollár, "Focal loss for dense object detection," in *Proc. ICCV 2017*. IEEE, 2017, pp. 2980–2988.

[99] G. Li, T. Liu, J. Nie, L. Guo, J. Chen, J. Zhu, W. Xia, A. Mara, S. Holley, and S. Wong, "Segmentation of touching cell nuclei using gradient flow tracking," *J. Microsc.*, vol. 231, no. 1, pp. 47–58, 2008.

[100] I. J. Goodfellow, J. Shlens, and C. Szegedy, "Explaining and harnessing adversarial examples," in *Proc. ICLR 2015*. Computational and Biological Learning Society, 2015.

[101] A. Fawzi, S. M. Moosavi Dezfooli, and P. Frossard, "The robustness of deep networks – a geometric perspective," *Signal Process. Mag.*, vol. 34, no. 6, pp. 50–62, 2017.

[102] C. G. Northcutt, A. Athalye, and J. Mueller, "Pervasive label errors in test sets destabilize machine learning benchmarks," *arXiv preprint arXiv:2103.14749*, 2021.

[103] L. Coelho, A. Shariff, and R. Murphy, "Nuclear segmentation in microscope cell images: A hand-segmented dataset and comparison of algorithms," in *Proc. ISBI 2009*. IEEE, 2009, pp. 518–521.

[104] M. Maška, V. Ulman, D. Svoboda, P. Matula, P. Matula, C. Ederra, A. Urbiola, T. España, S. Venkatesan, D. Balak *et al.*, "A benchmark for comparison of cell tracking algorithms," *Bioinform.*, vol. 30, no. 11, pp. 1609–1617, 2014.

[105] M. Maška, O. Daněk, S. Garasa, A. Rouzaut, A. Muñoz-Barrutia, and C. Ortiz-de Solorzano, "Segmentation and shape tracking of whole fluorescent cells based on the Chan–Vese model," *IEEE Trans. Med. Imag.*, vol. 32, no. 6, pp. 995–1006, 2013.

[106] A. A. Kalinin, A. Allyn-Feuer, A. Ade, G.-V. Fon, W. Meixner, D. Dilworth, R. Jeffrey, G. A. Higgins, G. Zheng, A. Creekmore *et al.*, "3D cell nuclear morphology: Microscopy imaging dataset and voxel-based morphometry classification results," in *Proc. CVPR 2018 Workshop*. IEEE, 2018, pp. 2272–2280.

[107] V. Ljosa, K. L. Sokolnicki, and A. E. Carpenter, "Annotated high-throughput microscopy image sets for validation," *Nat. Methods*, vol. 9, no. 7, 637, 2012.

[108] T. Lindeberg, "Feature detection with automatic scale selection," *Internat. J. Computer Vis.*, vol. 30, no. 2, pp. 79–116, 1998.

[109] L. Grady, "Random walks for image segmentation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 11, pp. 1768–1783, 2006.

[110] C. Chen, W. Wang, J. A. Ozolek, N. Lages, S. J. Altschuler, L. F. Wu, and G. K. Rohde, "A template matching approach for segmenting microscopy images," in *Proc. ISBI 2012*. IEEE, 2012, pp. 768–771.

[111] K. Magnusson and J. Jaldén, "A batch algorithm using iterative application of the viterbi algorithm to track cells and construct cell lineages," in *Proc. ISBI 2012*. IEEE, 2012, pp. 382–385.

[112] Y. Song, W. Cai, H. Huang, Y. Wang, D. D. Feng, and M. Chen, "Region-based progressive localization of cell nuclei in microscopic images with data adaptive modeling," *BMC Bioinform.*, vol. 14:173, no. 1, pp. 1–16, 2013.

[113] M. D. Zeiler and R. Fergus, "Visualizing and understanding convolutional networks," in *Proc. Eur. Conf. Comput. Vis.* Springer, 2014, pp. 818–833.

[114] P. Bamford, "Empirical comparison of cell segmentation algorithms using an annotated dataset," in *Proc. ICIP 2003*, vol. 2. IEEE, 2003, pp. II–1073–1076.

[115] M. Andersen, J. Dahl, and L. Vandenberghe, "CVXOPT: A Python package for convex optimization, version 1.1.4," http://cvxopt.org, 2011.

[116] C. Lorenz, I. Carlsen, T. Buzug, C. Fassnacht, and J. Weese, "Multi-scale line segmentation with automatic estimation of width, contrast and tangential direction in 2D and 3D medical images," in *Proc. CVRMed-MRCAS 1997*. Springer, 1997, pp. 233–242.

[117] P. Zhang, F. Wang, G. Teodoro, Y. Liang, M. Roy, D. Brat, and J. Kong, "Effective nuclei segmentation with sparse shape prior and dynamic occlusion constraint for glioblastoma pathology images," *J. Med. Imag.*, vol. 6, no. 1, 017502, 2019.

[118] T. Mollenhoff, E. Laude, M. Moeller, J. Lellmann, and D. Cremers, "Sublabel-accurate relaxation of nonconvex energies," in *Proc. CVPR 2016*. IEEE, 2016, pp. 3948–3956.

[119] E. Parzen, "On estimation of a probability density function and mode," *Ann. Math. Statistics*, vol. 33, no. 3, pp. 1065–1076, 1962.

[120] M. Andersen, J. Dahl, and L. Vandenberghe, "CVXOPT: A Python package for convex optimization, version 1.1.8," http://cvxopt.org, 2015.

[121] M. Andersen, J. Dahl, and L. Vandenberghe, "CVXOPT: A Python package for convex optimization, version 1.2.5," http://cvxopt.org, 2020.

[122] J. Funke, C. Zhang, T. Pietzsch, M. A. G. Ballester, and S. Saalfeld, "The candidate multi-cut for cell segmentation," in *Proc. ISBI 2018*, 2018, pp. 649–653.

[123] H. Bay, T. Tuytelaars, and L. Van Gool, "Surf: Speeded up robust features," in *Proc. ECCV 2006*, 2006, pp. 404–417.

[124] R. Franke, "Scattered data interpolation: tests of some methods," *Math. Comput.*, vol. 38, no. 157, pp. 181–200, 1982.

[125] N. Kumar, R. Verma, D. Anand, Y. Zhou, O. F. Onder, E. Tsougenis, H. Chen, P.-A. Heng, J. Li, Z. Hu *et al.*, "A multi-organ nucleus segmentation challenge," *IEEE Trans. Med. Imag.*, vol. 39, no. 5, pp. 1380–1391, May 2019.