Automatic approaches for microscopy imaging based on machine learning and spatial statistics

Ramin Norousi

München 2013

Automatic approaches for microscopy imaging based on machine learning and spatial statistics

Ramin Norousi

Dissertation
an der Fakultät für Mathematik, Informatik und Statistik
der Ludwig–Maximilians–Universität
München

vorgelegt von
Ramin Norousi
aus Teheran

Heidelberg, den 06.11.2013

Erstgutachter: Prof. Dr. Volker Schmid, LMU München
Zweitgutachter: Prof. Dr. Achim Tresch, Universität zu Köln
Drittgutachter: Prof. Dr. Christian Heumann, LMU München
Tag der Disputation: 07. Februar 2014

Contents

Abstract

One of the most frequent ways to interact with the surrounding environment occurs as a visual way. Hence imaging is a very common way in order to gain information and learn from the environment. Particularly in the field of cellular biology, imaging is applied in order to get an insight into the minute world of cellular complexes. As a result, in recent years many researches have focused on developing new suitable image processing approaches which have facilitates the extraction of meaningful quantitative information from image data sets. In spite of recent progress, but due to the huge data set of acquired images and the demand for increasing precision, digital image processing and statistical analysis are gaining more and more importance in this field.

There are still limitations in bioimaging techniques that are preventing sophisticated optical methods from reaching their full potential. For instance, in the *3D Electron Microscopy(3DEM)* process nearly all acquired images require manual post-processing to enhance the performance, which should be substitute by an automatic and reliable approach (dealt in Part I). Furthermore, the algorithms to localize individual fluorophores in 3D super-resolution microscopy data are still in their initial phase (discussed in Part II). In general, biologists currently lack automated and high throughput methods for quantitative global analysis of 3D gene structures.

This thesis focuses mainly on microscopy imaging approaches based on Machine Learning, statistical analysis and image processing in order to cope and improve the task of quantitative analysis of huge image data. The main task consists of building a novel paradigm for microscopy imaging processes which is able to work in an automatic, accurate and reliable way.

The specific contributions of this thesis can be summarized as follows:

- Substitution of the time-consuming, subjective and laborious task of manual post-picking in Cryo-EM process by a fully automatic particle post-picking routine based on Machine Learning methods (Part I).
- Quality enhancement of the 3D reconstruction image due to the high performance of automatically post-picking steps (Part I).
- Developing a full automatic tool for detecting subcellular objects in multichannel 3D Fluorescence images (Part II).
- Extension of known colocalization analysis by using spatial statistics in order to investigate the surrounding point distribution and enabling to analyze the colocalization in combination with statistical significance (Part II).

All introduced approaches are implemented and provided as toolboxes which are free available for research purposes.

Zusammenfassung

Einer der häufigsten Wege, mit dem Umfeld zu interagieren ist die visuelle Interaktion. Daher zählen die bildgebende Verfahren zu den sehr verbreiteten Ansätzen für die Informationsgewinnung und demzufolge das Lernen aus dem Umfeld. Speziell im Bereich der Zellkernbiologie werden bildgebende Verfahren eingesetzt, um Einblicke in die winzige Welt der Zellen zu verschaffen. Demzufolge haben sich viele Forschungsprojekte mit der Entwicklung von geeigneten Ansätzen für die Bildverarbeitung beschäftig. Diese Ansätze sollen dazu dienen, die Extraktion von bedeutungsvollen quantitativen Daten aus den Bildern zu ermöglichen. Aufgrund der großen Datenmenge der anfallenden Bilder und des Bedarfs an einer möglichst objektiven Untersuchung mit hoher Genauigkeit, haben die digitale Bildvearbeitung und die statistische Analyse viel an Bedeutung gewonnen.

Es gibt immer noch Einschränkungen bzgl. der Bio-Imaging Techniken, die uns daran hindern eine noch anspruchsvollere Methode für die Gewinnung der gesamten potenziellen Information zu erlangen. Beispielsweise im Bereich der *3D-Electron Mikroskopie* benötigen die aufgenommenen Bilder eine manuelle Nachbearbeitung um die Performanz und das Ergebnis zu optimieren. Dieser Schritt sollte anhand einer automatischen Routine ersetzt werden (Teil I). Des Weiteren befindet sich der Algorithmus für die Lokalisierung von fluoreszierende Proteine in hochauflösenden mikroskopischen 3D-Bilder in ihren Anfängen (Teil II). Im Allgemeinen fehlt es den Biologen gegenwärtig geeignete automatische Ansätze, die sie in die Lage versetzen mit einem hohen Durchsatz eine quantitative Analyse der Zellstrukturen durchführen zu können.

Die vorliegende Arbeit beschäftigt sich mit den Ansätzen aus den Bereichen des Maschinelles Lernen, der statistischen Analyse und der Bildverarbeitung, um die Aufgabe der quantitativen Analyse von großen Mengen an mikroskopischen Bildern zu bewältigen. Die Kernaufgabe besteht darin, neue Ansätze für die Bilder zu entwickeln, die in der Lage sind automatisch, präzise und zuverlässig zu analysieren.

Die wesentlichen Beiträge dieser Arbeit kann man wie folgt zusammenfassen:

- Ersetzung des zeitintensiven, subjektiven und aufwendigen Schrittes der manuellen Nachbearbeitung in der Cryo-EM Routine durch einen komplett automatisch ausführbaren Schritt, basierend auf der Methoden des Maschinelles Lernen (Teil I).
- Qualitätssteigrung der 3D-Rekonstruktion (Cryo-EM Prozess) durch den Einsatz einer automatischen Routine mit einem sehr hohen Durchsatz (Teil I).
- Entwicklung eines automatisch ausführbaren Tools für die Erfassung von zellulären Objekte in 3D Fluoreszenzbildern (Teil II).
- Erweiterung der bereits bekannten Kolokalisationsanalyse auf Basis der Ansätze aus dem Bereich der räumlichen Statistik, um die Punktumgebungen besser beschreiben zu können. Des Weiteren hat man die Möglichkeit das Ergebnis der Kolokalisation mit einer statistischen Signifikanz anzugeben (Teil II).

Alle beschriebenen Ansätze sind implementiert und stehen in Form von Softwarepakete für wissenschaftliche Zwecke zur Verfügung.

Danksagung

Für die Helden des Alltags, die ich im Laufe der Jahre kennengelernt habe. Ihr Wille der Überwindung von Hindernissen ist meine Inspiration. Durch sie habe ich immer neue Kraft gewonnen um weiter zu machen. Das ist Ihr Werk.

Mit großem Stolz stehe kurz vor der Verleihung meines Doktorgrades. Allerdings ist es mir sehr bewusst, dass ich diese Ehre und diesen Erfolg ohne eine Vielzahl von Menschen, die mich auf diesem Weg begleitet und unterstützt haben, nicht möglich wäre. Daher möchte ich an dieser Stelle bei vielen Personen, die mich bei der Erstellung dieser Arbeit sehr viel unterstützt haben, meinen tiefen Dank zum Ausdruck bringen.

Zuerst möchte ich mich bei meinem Doktorvater Volker Schmid bedanken, der mir mit seinem Fachwissen über die gesamte Zeit zur Seite stand. Er hat sich netterweise bereit erklärt mich als ein externer Promotionsstudent aufzunehmen und mich in dieser Form zu betreuen. Er stand mir jederzeit für konstruktive Gespräche zur Verfügung und mit seinem Fachwissen hat er immer meine Fragen gut beantworten können. Ohne ihn hätte ich niemals ein Licht am Ende der Doktorarbeit gesehen.

Besonders möchte ich mich bei Achim Tresch bedanken. Vor über vier Jahren hatte ich meinen ersten Kontakt mit ihm. Er hat schon vom Anfang an mich bestens unterstützt, sich viel Zeit für mich genommen und letztlich einen enorm wichtigen Beitrag für den Erfolg meiner Doktorarbeit geleistet. Er brachte mir sehr viel Geduld entgegen und sorgte mit wertvollen Ratschlägen, Ideen und Fachwissen für das Gelingen der Arbeit.

Ohne das Wissen und die Unterstützung meiner beiden Betreuern, ohne Ihre Ideen und Ihren Kritik wäre mein Forschungsprojekt niemals soweit gekommen.

Dafür bedanke ich mich recht herzlich bei diesen zwei wundervollen Betreuern.

Bedanken möchte ich auch besonders bei zwei Doktoranden, die mit mir bei den Projekten kooperiert und mich unterstützt haben. Stephan Wickles vom Genzentrum an der LMU, der mich während des ersten Projektes begleitet hat. Er hat mir die elektromikroskopische Bilder sowie die Bilder für den Trainingsdatensatz zur Verfügung gestellt und mein Tool schließlich auch evaluiert. Er hat die biologische Fragestellung aus dem Cryo-EM Bereich konkretisiert, die Anforderung präzise formuliert und mir tolle Feedbacks gegeben.

Bei meinem zweiten Projekt hat Christian Feller vom Institut für Molekularbiologie an der LMU mich beispiellos unterstützt. Er war ein Experte in seinem Gebiet, mit viel Know-How und Verständnis für die Bildverarbeitung und statistische Analysen. Wir haben uns regelmäßig ausgetauscht und uns getroffen. Die sehr konstruktiven Gesprächen führten zu interessante Ideen und erfolgreiche Schritte. Er war stets engagiert und stand auch an den Wochenenden für die Projektbesprechungen zur Verfügung.

Und nicht zuletzt danke ich meiner Familie, besonders meiner Mutter Mahin, die in jeglicher Hinsicht die Grundsteine für meinen Weg gelegt hat. Für Ihre liebevolle und herzliche Zuneigung zu mir bedanke ich mich ganz herzlich.

Ramin Norousi
Heidelberg, November 2013

Scope of this Work

This work deals with appropriate Machine Learning, image processing and statistical analysis applied on microscopy image data in order to avoid inaccurate and laborious manual interventions and hence to achieve automatically reliable and objective results. The focus of this work is to develop automatic approaches for picking particles and detecting subcellular objects in 3D microscopy image data. All introduced approaches are implemented and evaluated based on various real and artificial data sets. Furthermore the introduced tools are provided to the community as freely available toolboxes and packages for research purposes.

In detail, this thesis is organised as follows: Part I discusses the use of image processing and Machine Learning techniques for an automatic particle picking appliance in 3D Cryo-EM process, where Part II discusses object detection and spatial statistic approaches for 3D object detection and colocalization analysis in 3D multichannel Fluorescence images.

Furthermore, the individual parts are organized as follows: In Part I, Chapter 1 defines the 3D Cryo-electron Microscopy (Cryo-EM) process, followed by its main challenges, extended by a novel approach to solve described problems. Chapter 2 introduces the principle of available and engaged particle picking methods as well as a brief theory of Machine Learning in order to comprehend the methodology. Chapter 3 describes extensively the workflow of the implemented method. Chapter 4 describes the performed experiments and present the performance result of the implemented tool. Part I of this thesis is concluded with Chapter 5 which includes the main findings and advantages of the introduced method as well as a discussion of the main differences between our method and other developed approaches in this field.

In Part II, Chapter 6 introduces the concepts of Fluorescence Microscopy, colocalization analysis and the challenges in these fields. The extensive Chapter 7 describes most commonly used techniques and theory basics of image acquisition, spot detection, colocalization analysis and spatial point processes. In Chapter 8, we introduce our method step-by-step and describe the performed experiments and their results. Finally this part of the thesis is concluded in Chapter 9 with a summary of the main findings and a discussion of future works in this field.

Last but not least, the Appendix I and II provide the implementation details of both tools with computational details. It describes which inputs are required, which parameters should be set by the user and shows a sample run of them. Some further information such as time consumption and software requirements are also given.

It should be mentioned that some parts of this dissertation are based on published manuscript in the Journal of Structural Biology. In particular the majority part of Chapter 3, 4 and 5 from Part I was recently published in [Norousi et al., 2013]. This article contains contributions essentially by Achim Tresch and Volker Schmid. I performed all analyses, implemented the toolbox and wrote the main part of the article.

As mentioned all developed tools are freely available: First MATLAB package (*MAPPOS*) is available on both department (www.treschgroup.de/mappos.html) and private website (www.norousi.de). Second package (3D-OSCOS), relating to Part II, consists of MATLAB and *R* packages that both can be found on the website (www.norousi.de). The *R* package *bioimagetools* was created by Volker Schmid which was extended on the basis of the analysis in this work.

Part I

Particle Picking in 3D Cryo-EM

1. Introduction

> I never think of the future. It comes soon enough.
>
> Albert Einstein (In interview given aboard the liner Belgenland, New York, December 1930)

This chapter describes the process of the 3*D Electron Microscopy(3DEM)* after introducing the *Electron Microscopy (EM)* principle. In more details, Section 1.1 introduces the principles of Electron Microscpy and cryo-Electron Miroscopy. Further Section 1.2 describes the 3DEM process extensively and consequently in Section 1.3 the challenges of this process is characterized. Section 1.4 concludes this introductory chapter with an overview of our contributions in this work.

1.1 Cryo-Electron Microscopy (Cryo-EM)

It is widely recognized that the structure of a biological molecule is very crucial for its function [Helmuth et al., 2010]. Enormous efforts have been taken in solving the molecular basis of macromolecular complexes. Among all possible techniques in this field, the 3D reconstruction of biological specimens using *cryo-electron microscopy (cryo-EM)* is the most widely used technique. This method facilitates the visualization of three-dimensional macromolecular complexes in structural biology [Frank, 2006]. The principle of Cryo-Electron Microscope is introduced in the following section after clarifying the advantages of using Electron Microscopes.

In contrast to the conventional Light Microscopy, the Electron Microscopy is a very powerful technique that allows to obtain much more detailed information from specimen. The main distinguishing feature is that the EM uses a beam of electrons instead of photons to create an image of the specimen. The physical principle is the

same as light microscopy but it is able to work with a 10^5 times smaller wavelength which allows us to achieve a greater resolution (up to 100 times). Altogether the EM has a greater resolving power and is capable of a much higher magnification than a light microscope, allowing to see much smaller and finer details [Erni et al., 2009]. Two common types of electron microscopes can be distinguished, the *Scanning Electron Microscope (SEM)* and the *Transmission Electron Microscope (TEM)*. The TEM is the original form of electron microscopy [Zhao et al., 2006]

The *3DEM* (*3D Electron Microscopy*) is used in order to retrieve structural information from different biological macromolecular complexes, which is difficult by other known methods. Other common techniques like *negative staining* and *air-drying* (described in [Bozzola and Russell, 1999]) have two main drawbacks, firstly they can provide only information about the surface of molecule and secondly the resolution is not high enough. In contrast to them, the 3DEM uses samples embedded in vitrified ice reflecting the native and hydrate state [Nicholson and Glaeser R.M, 2001]. Furthermore the ability of Cryo-EM has been proven in investigating large biomolecules in sub nanometer resolution [Sorzano et al., 2009]. Therefore 3DEM has been a topic of interest in the field of structural biology for many years [Frank, 2006].

The *3DEM* method is based on Cryo-electron *micrographs*, which are captured by the Transmission Electron Microscope (TEM). As shown in Figure 1.1, a micrograph contains a number of low contrast two dimensional randomly oriented projections of biological molecules (referred to as *„particles“*) and further projections of none molecules i.e. ice, dust, contaminations or empty regions (referred to as *„non-particles“*). The 3DEM requires tens of thousands of projection that are frequently selected manually or semi-automatically from micrographs. The success of the 3DEM crucially depends on the number and the quality of the selected 2D particle images.

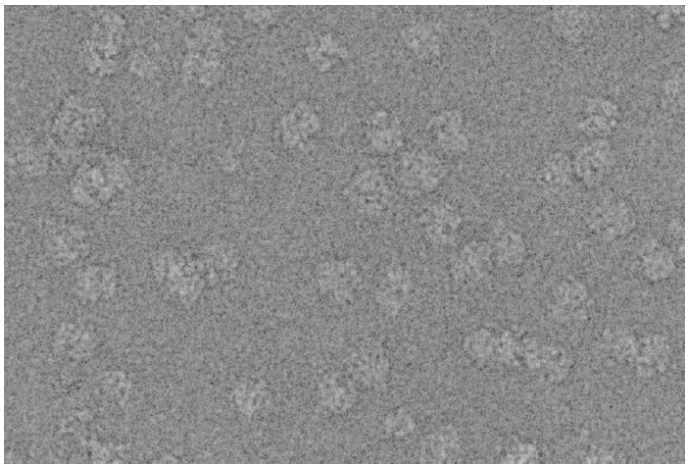


Figure 1.1: Sample micrograph of 70S ribosome

1.2 Process of 3D Electron Microscopy (3DEM)

As depicted in Figure 1.2, the process of 3DEM typically begins with acquiring images from a specimen by electron microscopy and draws to a close through a 3D reconstruction of the structure, based on alignment of acquired 2D images. It is described extensively in Frankś textbook [Frank, 2006]. For a good overview of different automatic particle selection algorithms, see, [Nicholson and Glaeser R.M, 2001] and [Zhu et al., 2004].

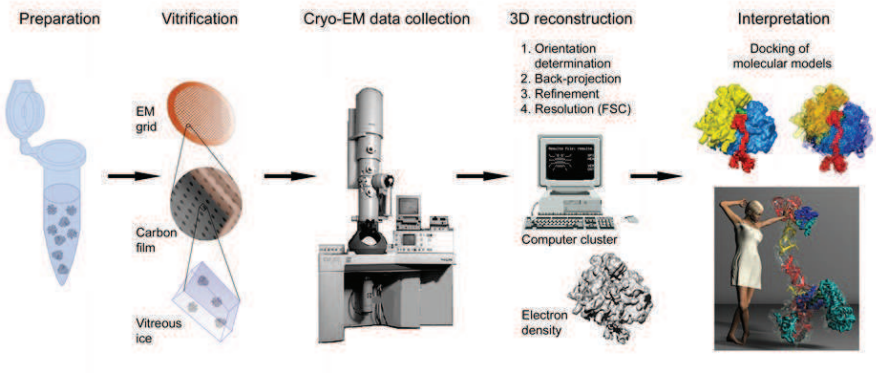


Figure 1.2: The process of Cryo-EM and single particle analysis [Frank, 2006].

Figure 1.2 depicts the process of 3D Cryo-EM, consisting of the following five steps:

1. **Preparation of molecular sample**

2. **Acquiring images from the specimen by electron microscopy**
   The image acquisition is executed after preparing biological molecule samples and their verification. Grayscale images (micrographs) can be recorded from the specimen using the lowest practical dose of electrons to avoid radiation damage to them.

3. **Automated particle picking using software tools**
   Finding and selecting of correct particles from the micrographs is one of the most crucial steps in this process.
   For huge data sets, software tools that inspect the micrograph and find particles based on various techniques (e.g. *cross-correlation* or *template matching*) are recommended (described in Section 2.1). These software tools determine the coordinates of putative particles which are windowed out for individual processing. The output of these software tools is a set of cropped images from micrographs. Analyzing this output set shows that the main part of the cropped images are correctly selected as particle and another part is a set of wrongly as particle selected images. One of the main objectives is to optimize this step in order to minimize the fraction of wrongly selected images.

4. **Manual particle post picking**
   As mentioned in step two, the output of particle picking tools contains some images which are wrongly selected (labeled) as particle, called *false-positives*. Due to the fraction of false positives in the output set, a manual post-picking process is required to remove it from the output set. Removing of false positives is very crucial, because otherwise it leads to artifacts in the 3D reconstruction. This step is the most time consuming and laborious task of the whole process of 3DEM which should be optimize or substitute by an automatic process. This stage is the focus of our project. The goal is to avoid the manual post picking step by establishing an automatic particle picking tool.

5. **Alignment and 3D reconstruction**
   The process of alignment is described extensively in [Frank, 2006]. During this process, the orientations of the randomly distributed particles have to be determined. Here the projection matching method will be used. In this method from a pre-existing reference, 2D reference projections are created and compared to the experimental particle images. Determine the angles of the 2D projection for a 3D reconstruction.
   There are 5 parameter, which have to be determined (3 Euler angles, 1 shift parameter in x-direction and 1 in y-direction). For correct reconstruction the three Euler angles, the in-plane translation and rotation is determined for every particle. The 3D reconstruction could be seen as reverse projection [Frank, 2006]. The 2D images represents the sum of the density values of the 3D object along the optical axis (Figure 1.3, left). That makes it possible to generate a three dimensional density map out of 2D projections if the projection angles for each particle are known (Figure 1.3, right).

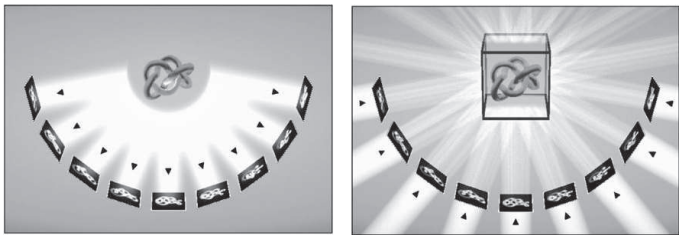


Figure 1.3: Image formation and reconstruction.
Left: Schematic showing the conversion of a 3D object to 2D projections.
Right: Schematic illustrating the back-projection of 2D images into a 3D object [Frank, 2006].

1.3 Challenges in 3DEM

Due to the bad *Signal to-Noise-Ratio (SNR)* in low-dose cryo-EM, several hundred thousand of 2D grayscale projections of a macromolecule (particles) are required [Woolford and Hankamer G. Ericksson, 2007]. Hence the 3DEM is dealing with a large amount of data, therefore the main objective of the 3DEM is to automate the steps of the 3DEM process as much as possible [Sorzano et al., 2009].

Further right from the early phase of 3DEM, it was noticed that manual particle picking from micrographs will become a labor-intensive bottleneck due to the following facts [Zhu et al., 2004]:

- A huge number (hundred thousand or even a million) of particle images are required for the 3D reconstruction.
- The micrographs are very noisy (typical SNR is 1) and they have a low contrast.
- Manually particle picking is a subjective task (various experts would interpret images differently).

Facing these facts, great efforts of fully and semi-automatic particle picking from low-dose electron micrographs during cryo-EM were made where SIGNATURE [Chen, 2007], SPIDER [Roseman, 2003] and EMAN2 ([Ludtke et al., 1999]) are very commonly used. A survey of the researches and techniques are described in [Zhu et al., 2004]. As mentioned before (Section 1.2) these tools are able to scan the micrograph and examine particles in accordance with defined criteria or templates. They select those areas of micrographs, which fulfill the criteria or match with defined templates and crop them from the micrographs. Their output is a set of cropped areas from micrographs containing a particle in random orientation. Figure 1.4 illustrates a sample of particle picking output where it can be recognized that it contains a fraction of false positives.

The main drawback of all particle picking methods is the typically large fraction of false positives, in the output set of these methods. The fraction of false positive images, depending on the method and the type of the specimen, lies between 10% up to 25% and leads to noisy 3D reconstruction [Zhu et al., 2004]. Hence for comparing the performance of these methods, the fraction of false positive rates are evaluated.

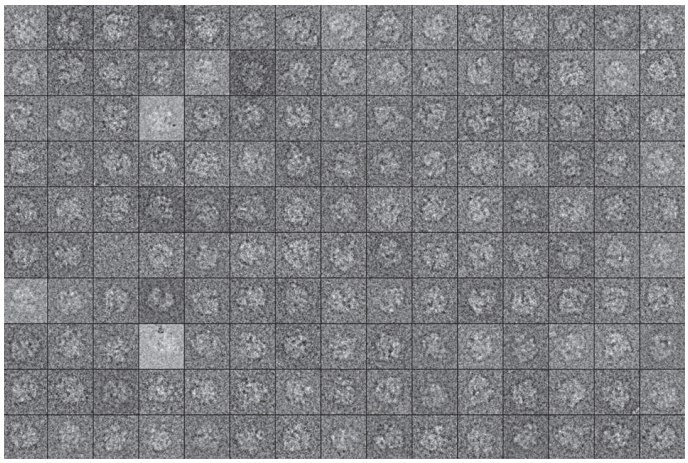


Figure 1.4: Sample particles selected from micrograph

To sum up, although automated particle picking methods are invaluable for processing large cryo-EM datasets, subsequent manual post-processing is still inevitable to eliminate non-particles. The manual post processing constitutes one major bottleneck for the next generation of electron microscopy and for high resolution reconstruction of unsymmetrical particles. Therefore an automatic approach to substitute the manual post picking is a significant contribution in this field which is the main focus of this work.

1.4 Our Contribution

Regarding the mentioned facts and challenges, an automatic workflow should be established to revise the output of automatic particle picking step in the sense that all cropped images from micrographs should be correctly classified into two classes particles and non-particles. The output of this workflow will be a set of images with minimum possible number of non-particle images.

Therefore instead of focusing on improvements in automated particle picking from micrographs, we propose a novel method to avoid the manual post processing step which is currently required due to the false positive rate. We introduce a method to investigate the output of particle picking methods and to classify them

automatically into two sets of particle and non-particle images.

Since the task of particle picking (step 2 in 1.2) is distinct from the task of discriminating particles and non-particles in a collection of individual boxed images, we suggest that both tasks should be addressed in individual steps. While elegant approaches exist for picking particles from micrographs or to reduce the time consumption of manual post picking step [Shaikh et al., 2008], we propose to subject the output of these automated particle picking methods to a specialized round of classification to separate particle images from non-particle images.

To achieve this task we established *MAPPOS* (**M**achine learning **A**lgorithm for **P**article **Pos**t-picking), a supervised discriminative post-picking method based on characteristic features calculated from a set of boxed images. First specific and essential features are learned from a provided training set by MAPPOS, after that MAPPOS is able to classify a set of new data into two groups of particles and non-particles, see Figure 1.5. The idea and workflow of Mappos is described in Section 3.1 in more details.

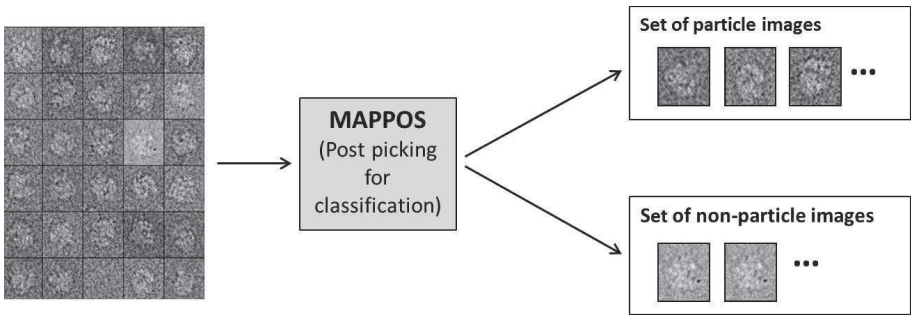


Figure 1.5: A rough survey of the classification idea of MAPPOS.

2. Theory Basics

> This morning I declined to write a popular article about the question „Can machines think?“.
> I told the editor that I thought the question as ill-posed and uninteresting as the question „Can submarines swim?“
>
> Edsger W. Dijkstra

This chapter deals first with the introduction of the *3DEM* process in general. Furthermore it describes the theory of *Machine Learning* which is used mainly in MAPPOS. An extensive explanation is beyond the scope of this thesis and we refer to the original publication for details (see [Frank, 2006] for 3DEM and [Hastie et al., 2009] for Machine Learning basics).

2.1 Particle Picking Methods in 3DEM process

Particle picking step consists of finding and cropping particle images in low-dose micrographs, which is one of the crucial steps in the 3DEM process. The goal is to improve this step in the sense that all particles should be picked with a minimum as possible fraction of non-particles.

Excellent particle picking methods have been developed ([Nicholson and Glaeser R.M, 2001]) and evaluated ([Zhu et al., 2004]).These proposed methods have met with varying degrees of success.

These methods can be divided, according to the used algorithm, into three categories:

- *Generative* approach.
- *Discriminative* approach.
- *Unsupervised* approach.

All of these approaches focus on the optimization of particle picking from micrographs. These methods yield sets of boxed images whose quality, as mentioned before, depends on both the signal-to-noise ratio of the micrograph and the particle picking method. These three categories can be described as follows:

Generative Method based on Template Matching

Template matching is a common technique for detecting and recognizing of patterns and is used in both signal and image processing. This method requires some *templates* (called also *references*) to detect particles in micrographs. Templates in different orientations can be generated from either a $3D$ reference structure or the average of a set of manually picked particles.

Generative approach measures the similarity between regions of the micrograph and the provided template using cross-correlation as a similarity score ([Chen, 2007], [Hall and Patwardhan A., 2004],[Huang, 2004],[Roseman, 2003]). Thus most of these methods are also called *template matching methods*.

As depicted in Figure 2.6, first a template T is required in order to search for desired objects. Furthermore, a window with the same size like the template (*search object* S) pasts over the micrograph. In each step the area under the window will be compared with the defined template. The cmparison is based on cross correlated [Turin, 1960]. If the value of the cross correlation is higher than a defined threshold, this area will be evaluated as a particle. Otherwise the area under the window is a non-particle. Hence in order to make a decision about the area under the search object, a suitable threshold should be assigned to discriminate between particle and non-particle.

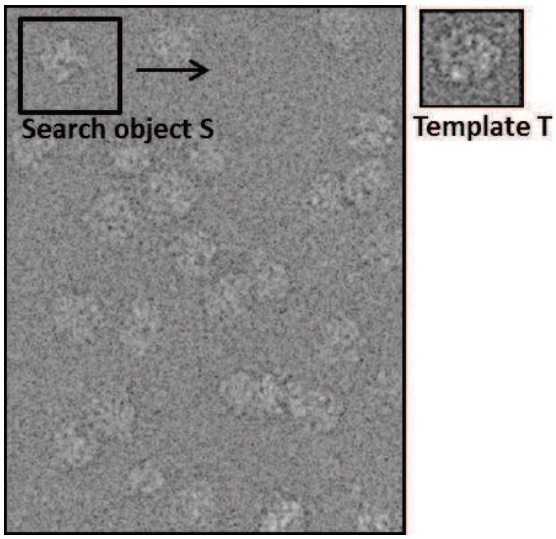


Figure 2.6: Searching for particles on micrograph based on templates.

Assuming that the search and the target objects are called S and T, respectively. The correlation coefficient of these two functions S_i and T_i over n points can be represented as follows [Roseman, 2003]:

$$C = \frac{1}{n} \sum_{i=1}^{n} \frac{(S_i - \overline{S})(T_i - \overline{T})}{\sigma_S \sigma_T},$$

where $\overline{S}$, $\overline{T}$ are the means of S_i and T_i and σ_S, σ_T are the standard deviations.

The SIGNATURE ([Chen, 2007]) is an established software in this field. This interactive software tool is used for picking particles from micrographs. The user can set the parameters as *local cross correlation function (lcf)*, *global cross correlation function (scf)* and pixel size. Furthermore some particle images are given as templates. Depending on the cross correlation value, as output a variety of small images cropped from the micrographs is given, which are evaluated as particle images (see Figure 1.4).

The main weakness of this technique is that its output consists of a high fraction of false positives. The reason for this is that some areas with an average intensity as the template, their calculated correlation value is high enough to evaluate them as a particle although it does not contain any particle [Zhu et al., 2004].

Discriminative Method based on Learning Algorithm

Discriminative methods require a set of training images instead of initial templates. The training set should contain both positive and negative samples of cropped images. According to these samples, a binary classifier is trained. This can be done either fully supervised using statistical learning ([Hall and Patwardhan A., 2004], [Mallick et al., 2004],[Volkmann, 2004]) or Machine Learning [Arbelaez et al., 2011] approaches, or in an iterative, supervised fashion [Sorzano et al., 2009] allowing the user to correct the algorithm during the training phase.

This method is a discriminative or learning-based approach, since the algorithm is trained and learned on a set of particles and non-particle features.

Unsupervised Method based Feature Recognition

The category of unsupervised approaches do not require a training set and work without any reference. Particles are automatically detected based on statistical measures and features that are extracted directly from the micrograph ([Adiga et al., 2005]; [Ogura and Sato C., 2004]; [Roseman, 2003]; [Voss et al., 2009]; [Woolford and Hankamer G. Ericksson, 2007]; [Zhu et al., 2004]).

The main characteristic of unsupervised methods is that in previous to the analysis, a set of suitable and discriminatory features of particles should be specified. Afterthat, the algorithm searches for these defined features in order to recognize them. For instance features like gray value, contours, lines and statistical features like moments, median of gray values are appropriate for discriminating particles from non-particles. Hence this approach consists of three phases: First the definition of a discriminative feature set, then the extraction of these features from an image and finally the recognition algorithms.

Feature based methods usually rely on a small set of features of images and unlike the template matching algorithms does not use a large number of pixels. But the main weakness of this method is due to the low contrast of the images, it would be a difficult task to extract distinctive features pertinent to a specimen [Zhu et al., 2004].

2.2 Principles of Machine Learning

One of the most fundamental field of artificial intelligence is the *Machine Learning* (or more general the *Pattern Recognition*). *Pattern Recognition* is the act of studying in raw data to find valuable and meaningful patterns. The recognized information and pattern can be used in order to make decisions about a new data set or to predict the future with a certain degree of likelihood [Duda et al., 2001]. This process has been crucial for our survival and we have involved highly sophisticated neural and cognitive systems for such tasks over the past tens of millions of years. For instance, humans perfom this task with remarkable ease. In early childhood we learn how to distinguish, for example, between apples and bananas.

However, unlike humans, in order to enable a computer to get the task of distinguishing objects or recognizing patterns in an automatical manner is a much more difficult challenge and often an ill-posed problem. Due to the need of learning from highly growing amounts of data („big data“), many researchers focus on pattern recognition as an essential research in the last decades. They investigated the way and the process of pattern recognition in human brains and tried to map it into a computer.

Several well developed methods and algorithms are provided in the field of Machine Learning which facilitates and supports our tasks in our daily life. Nowadays there are well-developed algorithms that are able to recognize a face, detect a fraud case, understand spoken words, read handwritten characters, identify DNA sequence and much more, however it is clear that reliable, accurate pattern recognition by a machine would be immensely useful in our daily life.

In this section, some of the most important concepts and methods of Machine Learning are introduced. Afterthat, some most commonly used approaches for measuring and evaluating the performance of Machine Learning methods are presented. The main part of this chapter is essentially based on [Hastie et al., 2009] and [Duda et al., 2001].

In general the learning task can be roughly classified in the following two types [Hastie et al., 2009]:

1. **Supervised Learning:**
   The *supervised learning* or *classification* is a two-step process, consisting of a *learning* (where a classification model is constructed based on a given training set) and a *classification* step (where the generated model is used to predict the class labels for new data).

   Thus the classification task requires a training set τ consisting of n labeled objects to learn about the objects and their labels. The main task is to generate a function (or model) based on the labeled training data. In other words, a supervised learning algorithm analyses the training data and produces a model, which is called a *classifier*. The classifier is used to predict class labels of objects for which the class labels are unknown (*prediction phase* or *test phase*).
   Since the MAPPOS is a supervised learning method, we are focused on the supervised learning algorithms which are described in more detail in the next section.

2. **Unsupervised Learning:**
   The *clustering* belongs to the unsupervised learning which is a tool for exploring the structure of data. It contrasts with supervised learning in the sense that the class label of each training object is not known. In addition the number of feasible classes also may not to be known.
   The core of cluster analysis is the process of grouping objects into clusters so that objects from the same cluster are similar and objects from different clusters are dissimilar. Objects can be described in terms of measurements (e.g. attributes, features) or by relationship with other objects (e.g. pairwise distance, similarity).

   Currently, there is considerable interest in better understanding gene functions in the biological processes of cells. A key step in the analysis of gene expression data is the detection of groups of genes that manifest similar expression patterns.

2.2.1 Supervised Learning Methods

As previously mentioned, the supervised learning task (classification), consisting of a *training*- and *test* phase, are described as follows [Hastie et al., 2009]:

Training phase: Assuming a set of training (learning) samples including n labeled observations is given:

$$\tau = (x_i, y_i)\,,\ i = 1, ..., n\ ,\quad x_i \in \mathbb{R}^{\mathrm{d}},$$

where each sample consists of a d-dimensional input (or *independent*) variable x_i that are called *input-features* and for each object the output (or *dependent*) which is the *class label* y_i is provided.

If the y_i is quantitative the prediction task is called *regression*, if it is qualitative (categorical, discrete) the prediction task is called *classification*. In case of categorical classification task with two possible output values $y_i \in \{0, 1\}$ or $y_i \in \{-1, 1\}$, the prediction task is a *binary classification*. MAPPOS is an example of a binary classifier.

During the training phase, the main task is to learn about the data and to estimate a good prediction model $\hat{f}$ of the output y_i from the training sample based on an algorithm a with $a(\ \cdot\ |\tau)$:

$$\hat{f}(x_i) = y_i + \epsilon$$

where the classification error is a random value ϵ with $E\,(\epsilon) = 0$.

Testing phase: In the testing or prediction phase the constructed model is used to predict the class label y_{new} for an unseen new object x_{new}:

$$y_{new} = \hat{f}(x_{new})$$

Among the supervised learning methods we introduce three frequently used algorithms that are deployed in MAPPOS. The theory of these Machine Learning algorithms are described extensively among others in textbooks of [Bishop, 2006], [Duda et al., 2001] and [Hastie et al., 2009].

1. Nearest-Neighbor Classifier

The *k-nearest neighbor (kNN)* algorithm [Cover and Hart, 1967] is one of the most intuitive supervised learners. The idea of kNN is based on the principle that samples with similar properties generally exist in close proximity than those with less similarity [Cover and Hart, 1967]. In most cases kNN is used in the initial phase of the study when there is a little or even no prior knowledge about the data.

Thus determining the location of the training data with a classification label, a new unclassified object x_{new} can be classified based on its proximity and distance to its classified next neighbors. From the x_{new} the k next neighbors are considered, and it will be classified to that class, which has the most objects among the k next neighbors as illustrated in Figure 2.7. The value of k is a usually rather small odd (to avoid tied votes) positive numbers and the correct classification of the neighbors is known a priori.

The distance to the neighbors of an unclassified object is determined by using a distance metric, for example the *Euclidean distance* or the *Manhattan distance*. A survey of different distance metrics for kNN classification can be found, for example, in [Weinberger et al., 2006].

Once we have some data and have chosen a suitable distance measurement, the most important setting is the choice of k. If k is too small, the classification can be affected by noise. As depicted in Figure 2.7 different values for k ($k = 3$ and $k = 5$) lead to various classification results. One possibility for choosing an appropriate value for k is to begin with a small value for k and check the performance and n next steps after increasing the value, we can check if increasing has a positive effect on the performance or not. We increase the value as long as the performance exhibits a positive development.

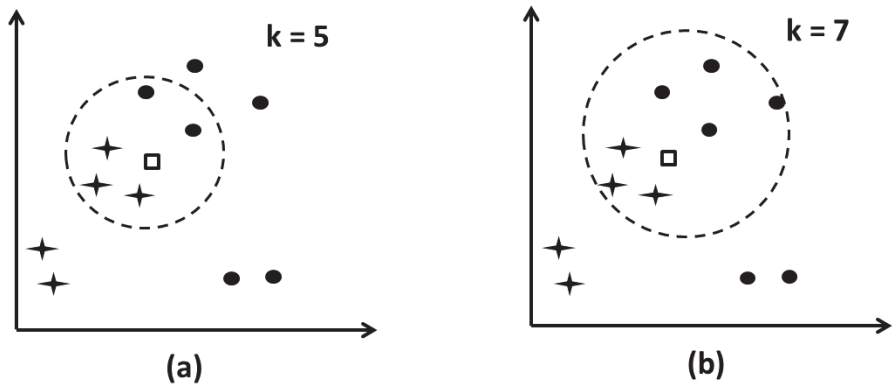


Figure 2.7: An illustration of kNN algorithm. From the unclassified query point x_{new} (depicted by a square) the next neighbors are searched for k=5 and k=7 in the left and right chart, respectively. Although both have the same proximity but as shown in (a) among k=5 next neighbors there are more objects from the star class and by increasing the k-value to 7 (b), there would be more objects from the other one.

2. Support Vector Machines

One of the mostly used and very successfully applied classification algorithms in Machine Learning is the *Support Vector Machines (SVMs)*. The theoretical foundations of this approach was given and introduced by Cortes and Vapnik [Vapnik, 1995] to the Machine Learning community.

The main idea of Support Vector Machine is that to solve the classification problem, it transforms training data into a higher dimension and within this new dimension, it searches for the optimal separating hyperplane (decision boundary) which separates the data into classes with the maximum margin. SVMs seeks the optimal separating hyperplane between two classes by maximizing the margin between the classes, hence, they are also referred to as *maximum margin classifiers* [Hastie et al., 2009].

A semifinite function is used for mapping of the original features (x, x') into a higher dimensional space [Hastie et al., 2009]:

$$(x, x') \mapsto k(x, x').$$

The function $k(.,.)$ is called the *kernel function* and uses Mercer's condition [Cristianini and Shawe-Taylor, 2000].

As depicted in Figure 2.8 for two-class, separable training data sets, there are lots of possible linear separators. Intuitively, a decision boundary drawn in the middle of the void between data items of the two classes (i.e. maximally far from any data point) seems better than one which approaches very close to examples of one or both classes. This distance from the decision surface to the closest data point determines the *margin* of the classifier. The nearest points of both classes to the decision surface are referred to as the *support vectors* (see Figure 2.8).

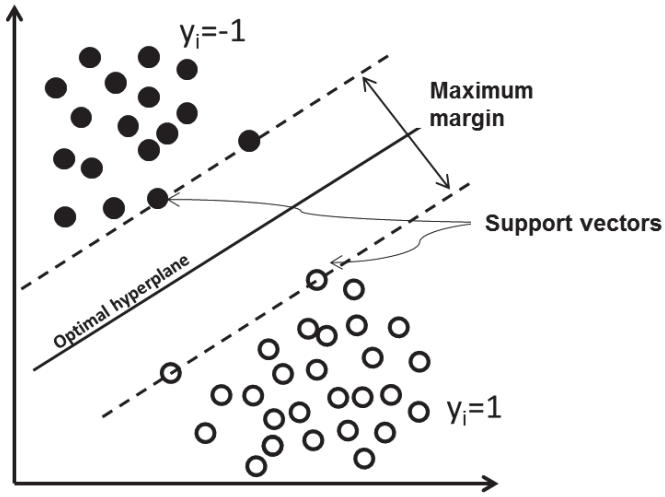


Figure 2.8: SVM classification algorithm applied on a two class problem. The best hyperplane which has the best separation quality is the solid line in the middle of the support vectors. The support vectors are the 4 points right up against the margin of the classifier.

3. Decision Trees

Decision trees (*classification trees* or *regression trees*) are tree-like models which support decision making in a simple form. They apply a *„divide-and-conquer“* approach to the problem of learning from a set of independent instances. All decision trees consist of several nodes that can be either an internal or a leaf node. Each internal node is a question on features that branches out (classifies) instances according to the answers. Each leaf node has a class label, determined by majority vote of training examples reaching that leaf. It is natural and intuitive to classify a pattern through a sequence of questions, in which the next question asked depends on the answer to the current question [Duda et al., 2001]. Another important advantage of decision trees which makes it much more applicable is that the results are very easy to interpret.

There are two popular types, one is the *„regression and classification“* called *CRT* developed by [Breiman, 1993] and its major competitor *ID3* with its later versions,*C4.5* and *C5.0* [Quinlan, 1986]. The main property of he $CART$ algorithm is the binary decision role which means that each decision leads to split the samples into two groups (binary) which are more similar. The *C4.5* algorithm which is the successor of $ID3$ is the popular in a series of classification tree methods and unlike $CART$ it also uses multiway splits.

It is important to note, that the most discriminating split is on the top of the decision tree. I.e. the most important discriminatory feature based on information retrieval theory is determined and placed next to the tree root. Further it is important to decide when the algorithm should stop splitting. In other words to weighing up between stopping splits and accept imperfect decisions or instead select another property and grow the tree further [Duda et al., 2001].

It is known, that trees have a high variance, so they benefit from the ensemble approach [Breiman, 1996]. The idea of ensemble is described in Section 2.2.3.

2.2.2 Classification Model Assessment

The previous section addressed the definition of three mostly used classification models, now one might well wondered which is the best classification model based on their performance and how can we compare different classification models based on their performance? In this section some ideas and techniques for calculating and estimating of classification error are introduced. In the context of performance assessment the *generalization performance* of a learning method is very important. The generalization performance described how the prediction model performs on a new „out od sample“ data. It guides the choice of learning method and provides us with a measure of the performance. The main part of this section is quoted from [Hastie et al., 2009].

Loss function: Considering a quantitative classification task with a given training set τ consisting of a set of an d-dimensional input variable x_i and their associated target variable y_i is given, as defined in Section 2.2.1. Further, a prediction model $\hat{f}(X)$ is constructed based on an estimation from a training which is constructed. For convenience we summarize all input variables x_i as X and all target variables y_i as Y.

Since the real target values y_i of each sample i is known, the performance of a generated model can be measured by comparing the real and assigned target label values based on the so-called *loss function* . The loss function for measuring errors between Y and $\hat{f}(X)$ is denoted by $L(Y, \hat{f}(X))$. Typically choices for error measuring are [Hastie et al., 2009]:

$$L(Y, \hat{f}(X)) = \begin{cases} (Y - \hat{f}(X))^2 & \text{, squared error} \\ |Y - \hat{f}(X)| & \text{, absolute error.} \end{cases}$$

Training error: The *training error* (i.e. training performance) is the average loss over the training sample [Hastie et al., 2009]:

$$\bar{E}_\tau = \frac{1}{n} \sum_{i=1}^{n} L(y_i, \hat{f}(x_i))$$

Unfortunately training error is not a good measurement of the model performance. The training error decreases with model complexity and if the complexity is

high enough the training error moves towards zero and thus it is overfitted to the training data. In this case the model would have a low generalization performance.

Test error: Based on the loss function the *test error* (i.e. test performance) also referred to as *generalization error* is the prediction error over an independent test sample Λ [Hastie et al., 2009]:

$$\bar{E}_{\Lambda} = E[L(Y, \hat{f}(X)|\tau]$$

where both X and $Y$are drawn randomly from their joint distribution (population). For measuring the quality of the models the test error is decisive.

Errors in Binary Classifications

In case of a binary classification task which is a particular case of quantitative classification problem, the measuring of the loss function is slightly is much more convenient. Since in a binary classification problem, the possible output classes has only two possible characteristics (positive or negative). Therefore, a learning algorithm can make two types of errors (FP: **F**alse **P**ositive, FN: **F**alse **N**egative). Further a correct classification is referred to as trues (TP: **T**rue **P**ositive, TN: **T**rue **N**egative).

For visualization of both error types, the confusion matrix is used. The confusion matrix defines the four possible outcomes of a classification in a 2×2 table, as shown in 2.1. The columns tabulate the number of samples in the actual class and the rows of the predicted class. The two classes are referred to as the positive class P (or the class of interest) and the negative class NP (of the class uninterested).

In addition, a cost can be associated with each type of error. We can define if both errors should be penalized equally by 1 or associate the errors with different values. E.g. to classify a patient with cancer as healthy should be penalized much higher than the inverted case (classifying a healthy person as cancer patient). This setting of miss-classification cost is defined in a so called *cost matrix*.

Table 2.1: Confusion matrix with four values that reports prediction performance: The columns tabulate the actual class and the rows of the predicted class. E.g. the value FN is derived when the actual class label is positive and the predicted class is negative.

	Actual class(+)	**Actual class(-)**	**TOTAL**
Predicted class(+)	True positive (TP)	False positive (FP)	Total predicted positives (P')
Predicted class(-)	False negative (FN)	True negative (TN)	Total predicted negatives (N')
TOTAL	Total actual positive (P)	Total actual negative (N)	

There is a set of explicitly defined and widely used performance metrics within the field of Machine Learning to evaluate the binary classification models. The mostly used metrics are: *Sensitivity*, *Specifity*, *Accuracy*, *Precision* and *Recall* [Hastie et al., 2009].

Two principally important measures for validity and performance of binary classification models are the *Sensitivity*, which is the fraction of correctly identifying positives (TP) to total positives in the population ($P = FN + TP$),

$$Sensitivity = \frac{TP}{TP + FN} = \frac{TP}{P}$$

and the *Specifity* that quantifies how well a binary classification model correctly identifies the negative cases by fraction of true negatives (TN) over the total negatives ($N = FP + TN$):

$$Specifity = \frac{TN}{TN + FP} = \frac{TN}{N}.$$

Receiver Operating Characteristic (ROC) & Area Under the ROC Curve (AUC)

If a large number of trials by varying the discriminatory threshold is possible, we can determine the performance of each threshold experimentally. The tradeoffs between the hit and false alarm rates, in particular the sensitivity and specifity can be determined. This tradeoff can be visualized in a two-dimensional plot by the so called *receiver operating characteristic (ROC)* [Fawcett, 2006]. The ROC, as is shown in Figure 2.9, is the graphical plot of the sensitivity versus the (1-specifity) for a binary classifier as its discrimination threshold is varied. In the ROC curve, the sensitivity is plotted on the y-axis and 1-specifity on the x-axis, which are referred to as the *true positive rate (TPR)* and *false positive rate (FPR)*, respectively.

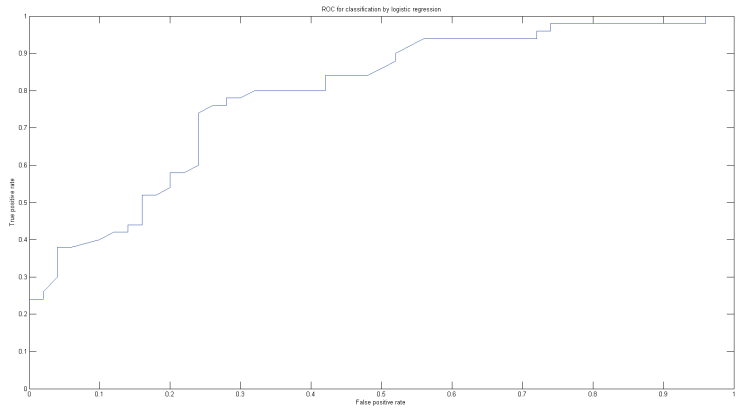


Figure 2.9: An illustration of the ROC-curve based on Fisheriris data from [K. Bache and M. Lichman, 2013]

Another very important measure is the *AUC (area under the ROC curve)* which serves as a useful measure to summarize the overall performance of a classifier. The ROC-curve offers the opportunity to calculate the specifity at a fixed sensitivity level and vice versa [Langlois, 2011]. The AUC ranges between 0.5 (equivalent to random guessing) to 1 (perfect classification). Furthermore, if the dataset is balanced, the next metric which is often used is summarizing both sensitivity and specifity in a single metric *the accuracy* defined by [Hastie et al., 2009]:

$$Accuracy = \frac{TP + TN}{FN + TP + TN + FN} = \frac{TP + TN}{P + N}$$

The *precision* or *positive predicted value (PPV)* is the proportion of all as positive classified objects which are correct and *Recall* is the fraction of positive objects which are classified as positive

$$Precision = \frac{TP}{TP + FP}$$

$$Recall = \frac{TP}{TP + FN}$$

Estimating the Prediction Error

In the last section some basic measurements like the loss function and errors in binary classification are introduced. These are essential for dealing with Machine Learning methods in order to understand and evaluate the classification task. Based on these definitions three most commonly used approaches for measuring the performance are described in following sections. The idea of these approaches is to measure the performance based on the provided training set and thus to estimate the prediction error, since the labels of the training set are known in advance. Three common ideas of performance measurement in Machine Learning are *Cross validation*, *K-fold cross validation* and *Bootstrap* which are described below [Hastie et al., 2009]:

1. Cross Validation

Evaluation of a classifier and estimation by its prediction error requires splitting aa a part of the training set, so-called *validation set* to assess the performance of the prediction model based on this set. As shown in Figure 2.10, some data (e.g.≈ 20%) is removed from the original training set which forms the validation set. After that a classifier is constructed based on the remain training set (e.g.≈ 80%). The constructed classifier is evaluated on the validation set.

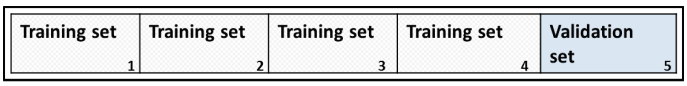


Figure 2.10: Illustration of Cross Validation by partitioning the training set.
As sample the number of subsets $k = 5$.

Cross-validation uses probably the simplest and most widely used method for estimating prediction error . It has two drawbacks, firstly it is sensitive to the choice

of data in the validation set and secondly it can be used only when enough sample data is provided [Hastie et al., 2009].

2. K-fold Cross Validation

As in many cases the provided data is rare, an extension of cross-validation referred to *k-fold-Cross-Validation* is used. It is a technique that allows us to make more efficient use of the data we have.

Similar to cross validation, two parts of training and validation set are used, but in more than one round. The validation consists of k rounds and in each round the K-fold cross-validation splits the data into k roughly equally sized subsets, as illustrated in Figure 2.11. In each round the model is train based on $k-1$ subsets and validated on the remaining validation set. Averaging from the resulting k loss values gives us our final loss value.

If we lack of relevant problem-specific knowledge, cross validation methods could be used to select a classification method empirically [Hastie et al., 2009].

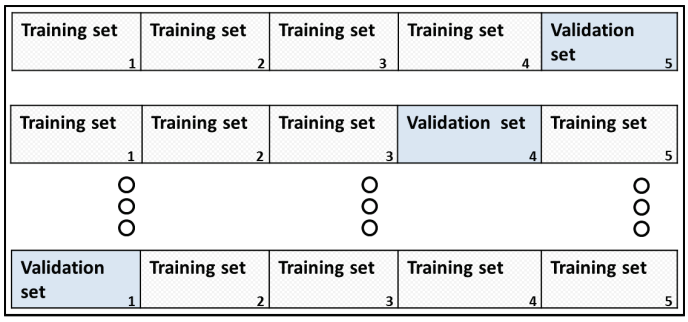


Figure 2.11: An example of K-fold Cross-Validation ($k = 5$).
The provided data is subdivided into $k = 5$ sets. A classifier is constructed using four sets of training and will be evaluated on remaining set (the validation set). It consists of k rounds of training and validation of classifiers. In each round the performance is measured and in the end the error rates of all k rounds are averaged. This average value of error rates serves as a performance measure.

3. Bootstrap

Bootstrap is the next widely used technique for estimating prediction error with a difference in how to create the individual training and validation samples.

Unlike the prediction error estimation by cross-validation, in the *bootstrap method* the given training set samples are selected randomly and uniformly with replacement to form the training and validation sets. The samples selected by cross validation are dependent and it is not possible to use it in a more randomly manner (e.g. using some samples randomly several times in various subsets).

A „bootstrap“ data set is created by randomly selecting n samples from the training set τ. Because τ itself contains n points, it is very likely that some of the original data samples will occur more than once in this sample.

The basic idea is to randomly draw independently d learning sets with replacement from the original learning set τ:

$$\tau^1 = \left\{z_1^1, \cdots, z_n^1\right\}$$

$$\vdots$$

$$\tau^d = \left\{z_1^B, \cdots, z_n^B\right\}$$

Suppose a data set with d samples is given. The data set is sampled d times, with replacement, resulting in a bootstrap sample or training set of d samples. It is very likely that some of the original data samples will occur more than once in this sample.

Assume we try this out several times. As it turns out, on average, $63,2\%$ of the original data samples[1] will end up in the bootstrap sample, and the remaining $36,8\%$ will form the validation set.

[1]The probability for each sample to be chosen is $1/d$ and the probability for its counterpart is $(1 - 1/d)$. We have to select d times and since the selecting is with replacement and therefor independent, the probability taht a sample will not be chosen in this whole time is $(1 - 1/d)^d$. If d is large, the probability approaches $e^{-1} = 0,368 = 36,8\%$ and thus the counterpart $1 - 0,368 = 63,2\%$.

2.2.3 Classification Ensemble

It is widely recognized that combining multiple classification or regression models typically provides better results compared to using a single, well-tuned model [Duda et al., 2001]. The idea behind ensemble methods can be compared to situations in real life. In fact, for a critical decision, asking several experts about their opinion and combining them generally leads to a better decision than asking just one expert.

An ensemble classifier consists of a set of independent classification algorithms for the identical classification problem. The decisions of its individual members are combined to one final prediction of the ensemble. The main task after collecting all decisions is how can a final decision be derived from a set of decision which can be different. A simple approach is to make a final decision based on the majority vote of individual decisions [Duda et al., 2001].

The idea of combining (ensemble) several decisions was first introduced in the neural networks community as it was discovered, that a combination of several Neural Networks can improve the model accuracy [Hansen, 1990].

Building ensemble of models is a common way to improve classification models in terms of stability and accuracy. In order to construct of an ensemble different classifiers are required. A very common approach to construct various classifiers based on a single training data is introduced in the next section [Hastie et al., 2009].

Bagging: (Bootstrap aggregating)

The name Bagging is derived from „**b**ootsrap **agg**regat**ing**“ and like bootstrap it uses multiple versions of training subsets, which are created by drawing randomly from the training set τ with replacement.

First the process of *bootstrap* is applied to generate d different subsets of the training set (bootstrap sample) where some samples could appear in more than one subset. Each subset is used to generate a classification model and the d classification models are fitted using the above d bootstrap samples and combined by averaging the output (for regression) or voting (for classification). Thus Bagging uses these created bootstrap data sets to train a different component classifier and the final classification decision is based on the vote of each classifier component [Duda et al., 2001].

Bagging was first proposed by [Breiman, 1996] in order to improve the classification by combining classifiers trained on randomly generated subsets of the entire training sets.

2.2.4 Challenges in Machine Learning

The main challenge in generating a classification model is to get a balance between performing well on the training set and having good generalization power. In this context there are two important characteristics which should be taken into account [Hastie et al., 2009]:

Overfitting:

Overfitting occurs when a generated classification model based on the training set is very complex and allows an almost perfect classification performance, but doesn't perform well on an out-of-sample data. Parameter tuning in the classification phase enhances the risk of overfitting. Hence the central goal of the classification approach is the generalization ability and avoiding *overfitting*. One possibility is to generate a method with as few external parameters as possible [Bishop, 2006].

The classification algorithm should be tried to learn from the true patterns (regularities) in the data as much as possible and ignore the irregularities or noise. In order to measure this, the generalization performance is a good indicator for measuring the performance [Hastie et al., 2009].

Bias-Variance Decomposition:

It should be taken into acount that the complexer the model, the less the generalization property of the model. Therefore in the learning phase it should be specified how strong the model should be fitted to the training set. Thus an essential challenge is to find the right balance between model complexity and generalization in order to avoid overfitting as depicted in Figure 2.12. This correlation is known in the Machine Learning as the *bias-variance decomposition*.

The variance correlates with instability of a model based on training set. One can check the instability (or variance) of the model by testing if small changes in the training set lead to generating completely other model or not? If so, then the model is instable and has a high variance. Thus models with too much flexibility that are rather complex, have a high variance while models with little flexibility, such as linear polynom functions have a little variance and a high bias [Duda et al., 2001].

In general the goal is to find the best model complexity which are not to simple that is unable to explain the differences between two classes and yet not so complex as to have pooor property [Hastie et al., 2009].

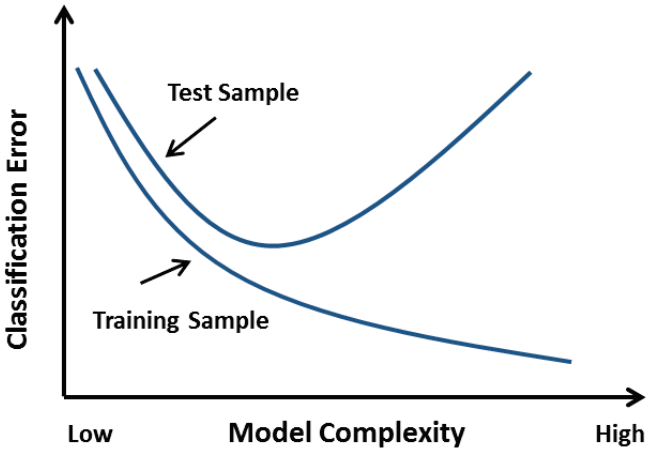


Figure 2.12: The balance between model complexity and generalization ability of the classification model. The highly the complexity of a classifier the lower is the error on the training sample, but the error on the test sample would increase for complex classifiers. Trading of the goodness of the fitting against the complexity of the model. [Duda et al., 2001].

3. Material and Methods

„The whole is more than the sum of its parts.“

Aristotles, Metaphysica

This chapter describes the developed methodology and introduces the approach to validate it using different real and artificial image data sets. It should be noted, that the content of this chapter is based on [Norousi et al., 2013].

3.1 Workflow of MAPPOS

MAPPOS is applied to substitute the third step of the 3DEM process (manual particle post picking) as described in Section 1.2. It is a supervised, generative and automatic approach based on a provided training set. It learns from characteristic features of the training set and constructs a classifier. Subsequently, this classifier can be applied to sort a new unclassified dataset automatically.

Referring to the standard workflow for a classification task [Bishop, 2006], as depicted in Figure 3.13, MAPPOS can be divided into a *learning phase*, followed by a *prediction phase* (particle detection) . The method relies on the availability of a relatively small set of training sample images which have been labeled manually as particles (+) or non-particles (-). This training set should contain a few hundred sample classified images which contain an approximately balanced number of samples of both image types. From each training sample, a vector of numerical features is extracted. A feature is a one-dimensional statistic that is calculated from a sample object. Together with the labels, the feature vectors serve as input to the learning algorithm.

We evaluated the performance of several algorithms and decided to use an ensemble of several classification models. The result of the learning phase is a binary classifier C which during the prediction phase assigns a binary label (+/-) to each image from a set of new, unclassified images.

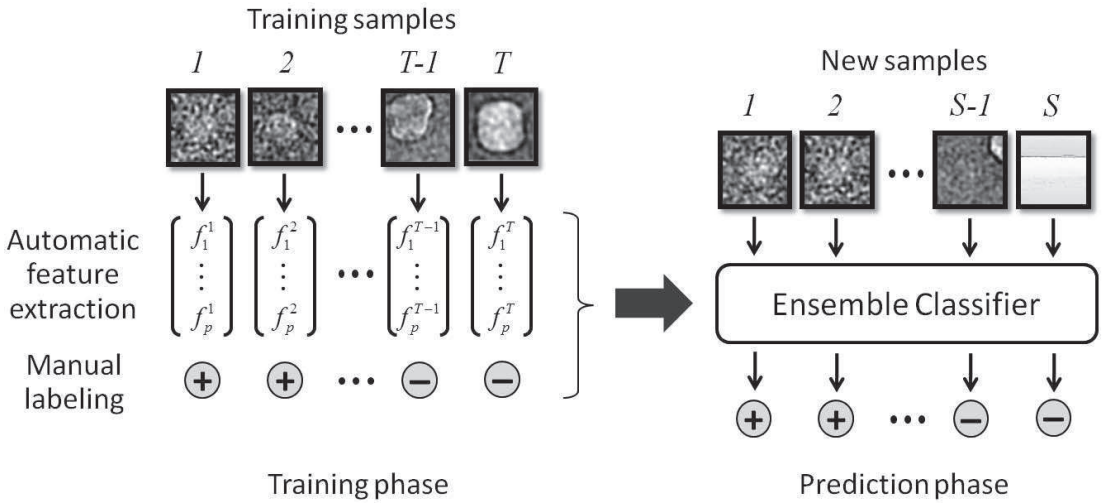


Figure 3.13: Workflow of MAPPOS.
A training dataset is created by manual classification of T sample images (typically, $T \approx 1,000$) as particles or non-particles labeled (+) or (-), respectively. During the training phase, p discriminatory features $(f_i^p, \cdots, f_p^j)$ are automatically extracted from each sample image j. The feature matrix (f_k^j) is used in combination with the sample labels to train an ensemble classifier. The ensemble classifier is used during the prediction phase to efficiently classify all of the S images (typically, $S \approx 10^5 - 10^6 \succ T$) of the complete dataset.

A summary of the process is that first of all a training set should be provided which includes a balanced number of particles and non-particles. Further appropriate features with good discriminatory properties are required. Based on these objects a classifier can be constructed. These three main steps are described in next sections.

3.1.1 Construction of a Training Set

We suggest running MAPPOS with a hand-picked training set of 500 particle images and 500 non-particle images. If artifacts of different types are existing (e.g. those mentioned in Fig. 3.16), it is advisable to choose the non-particles evenly from each type.

It is good practice to cross-check the final output of MAPPOS by eye to ensure a sufficiently high specificity of the selection procedure. In case it needs to be increased, the initial training set should be extended by another set of hand-picked non-particle images, say 500. This process can be iterated, however this was never necessary in our applications.

3.1.2 Determining of Discriminatory Features

The success of MAPPOS crucially depends on the definition of meaningful features, which as an ensemble, have a good discriminatory power. Therefore we set out to develop a fast and reliable classification method for post-picking of boxed cryo-EM images into particles and non-particles. To achieve a high robustness we avoid any user-adjustable parameters, thereby minimizing the risk of over-fitting.

A number of discriminatory features for applicability to this problem were tested and seven well-performing features that constitute the input to MAPPOS were identified:

- Location and scale (mean, variance)
- The (0%, 10%, 50% ,90% ,100%)-quantiles of the pixel intensity distribution
- Number of foreground pixels after binarisation using Otsu's thresholding [Otsu, 1979]
- Number of edges counted after Canny edge detection [Canny, 1986] (see Appendix 1)
- Radially weighted average intensity
- Phase symmetry / blob detection,
- Dark dot dispersion.

where the last three discriminatory features are described in more detail in the Materials and Methods section.

The discriminatory power of a single (continuous) feature is assessed by a ROC-curve ([Bradley, 1997]; [Fawcett, 2006]), see also section 2.3. Based on this criterion, we identified the following promising features (AUC values are given in brackets):

Radially Weighted Average Intensity (0.83): The radially weighted average intensity is calculated as a weighted sum of the pixel intensities, the weights being inversely proportional to their Euclidean distance from the center of the image. This statistic measures the centrality of the bright pixels, which for particle images should exceed that of non-particle images.

Phase Symmetry / Blob Detection (0.94): Blob detection [Kovesi, 1997] is based on the notion of phase symmetry, a contrast- and rotation invariant measure of local symmetry at each point of an image. Phase symmetry recurs on a 2D Wavelet transformation that extracts local frequency information [Morlet, 1982]. We apply the phase symmetry transformation with standard parameter settings as in [Kovesi, 1997]. The transformed image is binarized using Otsu's thresholding [Otsu, 1979]. Afterwards, locally symmetric areas (blobs) mainly occurring in non-particle images can be counted. We report the relative frequency of 1's in the binarized picture as a feature (Figure 3.14).

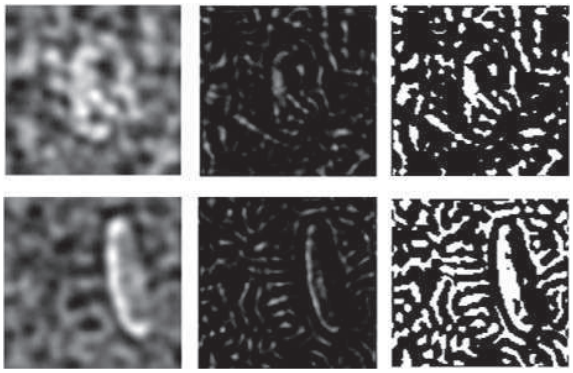


Figure 3.14: Phase symmetry transformation and binarization.
A particle image (top row) and a non-particle image (bottom row) are depicted in their original states (left column), after phase symmetry transformation (middle column), and after binarization (right column). The non-particle image contains an overall higher degree of symmetry and contains more white pixels after binarization.

Dark Dot Dispersion (0.86): We noticed that in particle images, the „dark dots“ are distributed more evenly across the image than for non-particle images. After convolution of the image with a 2-dimensional symmetric Gaussian kernel, dark dots are defined as connected regions of intensity less than the 5% quantile of the overall intensity values. The center of a dark dot is calculated as the mean of its pixel coordinates. The dark dot dispersion of an image is defined as the variance (the mean squared Euclidean distance) of its centers. Further helpful features were the (0%,10%,50%,90%,100%)-quantiles of the pixel intensity distribution of an image, the number of foreground pixels after binarization, and the number of edges counted after Canny edge detection [Canny, 1986].

3.1.3 Construction of an Classifier Ensemble

MAPPOS uses ensemble learning principles to construct ensemble classifiers from a set of individual classifiers. An ensemble classifier, as described in 2.2.3, consists of a set of k elementary independent classifiers $(C_1, ..., C_k)$ for an identical classification problem. The k binary predictions are combined to one final prediction by choosing the prediction made by the majority vote of the individual classifiers [Hansen, 1990]. An ensemble classifier generally yields an improved classification accuracy compared to each individual classifier [Duda et al., 2001].

We implemented in MATLAB the approach described in [Wichard, 2006] where bootstrap aggregating („*Bagging*“) approach [Breiman, 1996] for the construction of an ensemble is used. As showed in Figure 3.15, prior to the learning procedure -to later assess the performance of the final ensemble classifier as unbiased as possible- a validation set comprising 10% of the training data is held aside. Once the final classifier is constructed, its performance is evaluated on the validation set. The k elementary classifiers are iteratively selected out of a basic variety of classifiers and parameter settings. To that end, the remaining 90% of the training data are randomly split 5 times by subsampling an inner training set (80%) and an evaluation set (20%).

During our research several classifier models were used and analyzed on their performance. We tested among others *Linear discriminant analysis* [Mika et al.,], *Decision trees* [Quinlan, 1986], *Support vector machines (SVM)* [Vapnik, 1995], and *N-nearest-neighbors* [Cover and Hart P., 1967]. All models belong to the well-established collection of machine learning algorithms for classification tasks, details can be found in the textbook Hasti et al [Hastie et al., 2009] and Duda et al. [Duda et al., 2001].

We carried out an investigation with all classifier models assigned with various randomly generated parameters to cover diverse models with different setting parameters. Each candidate classifier is equipped with parameters randomly drawn from an appropriate range. The candidate classifiers are trained 5 times using the 5 inner training sets, respectively. Subsequently, they are applied to the corresponding 5 evaluation sets, and the classifier which performs best is added to the classifier ensemble. This process is iterated until no improvements can be made by the addition of another classifier [Wichard, 2006]. In our case, the final classification ensemble

contained 21 elementary classifiers.

After execution of k rounds and generating an ensemble with k classifier models, in the last step the performance of the automatic generated classifier ensemble can be measured by the validation set which was unseen during the CV rounds.

Assuming there are k several different classifier models $f_i(x)$ with associated weights w_i, then the ensemble classifier is determined by averaging the output of single models:

$$\hat{f}(x) = \sum_{i=1}^{k} w_i f_i(x).$$

The model weights w_i sum to one and there are several suggestions concerning the choice of the model weights. We decided to use uniform weights with $w_i = 1/k$.

3.2 Validation of MAPPOS

The Performance of MAPPOS was assessed on simulated and real data of the *E. coli ribosome*. For the simulated data, the true labels of the images were known.

Standard performance measures were calculated for 2×2 contingency tables of true vs. predicted labels (for their definition, see Table 2.1). For the real data, the manual classification was taken as a true label (gold standard). We additionally assessed the quality of the electron density map after 3D-reconstruction. In order to be self-contained and to avoid misunderstanding, we introduce some performance scores that have been proposed in [Langlois, 2011] for the comparison of particle picking methods.

As our classification problem is a binary decision (either particle or non-particle) we have two possibilities for the prediction (true or false). Thus each predicted value is either TP, TN, FP or FN. These numbers are conveniently displayed in a confusion table (Table 2.1 in Section 2.2.2)

Our algorithm was designed to search for contaminations in cryo-EM data sets. This implies that the *true negative (TN)* of the classification includes all non-particles which were correctly detected by our algorithm. The *false negative (FN)* therefore contains ribosomal particles which were incorrectly classified as contamination.

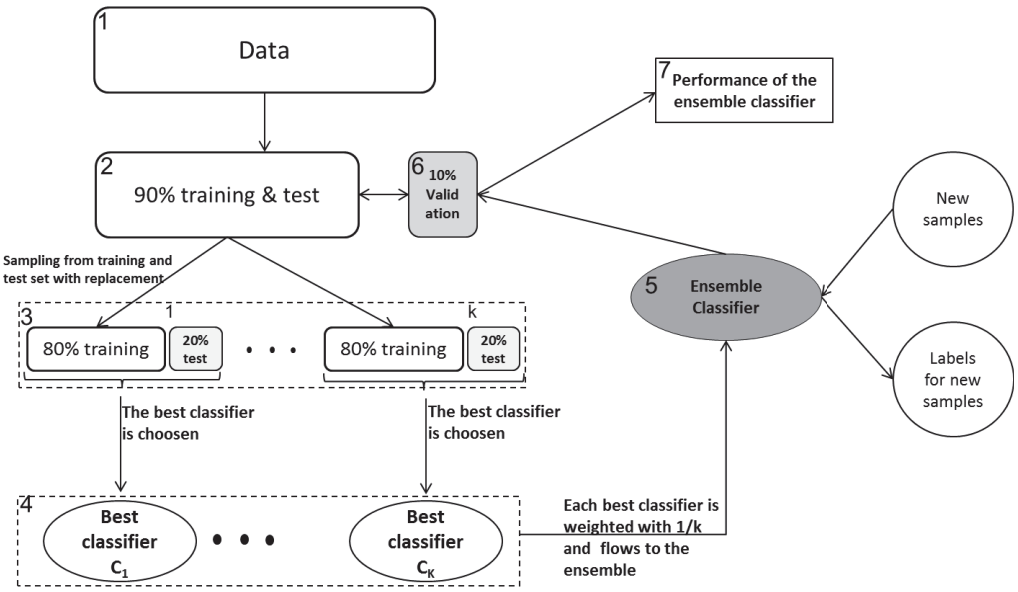


Figure 3.15: The process of the classifier ensemble construction. The steps of the process are consecutively numbered. First from the provided data a part of 10% is held aside for the validation of the ensemble in the end. The remaining data is randomly split k times by subsampling with replacement into a training and test set. For each subsample some classifier candidates are trained and tested on the same data. The classifier with the best performance is chosen and weighted by $1/k$ to construct the classifier ensemble. The constructed classifier can finally evaluated on the validation set. This ensemble is additionally can be used to classify a set of new samples.

As performance measures, we report *Sensitivity*, *Specifity*, *Positive predicted value (PPV)* and *accuracy* (see section 2.2.2). The choosing criteria for the best classifier is based on the accuracy measure. We were trying to maximize the PPV and the specificity while accepting a lower sensitivity.

3.2.1 Validation based on Artificial Data

We used a simulation environment for the generation of realistic particle and non-particle images. With this controlled environment we had a tool at hand to accurately quantify the quality of post-picking classifications and used it to assess the performance of MAPPOS and compare it to the manual performances of experienced

researchers (referred to as *„human experts“*).

We generated 21,922 windowed images with a particle/non particle ratio of 5/1, which is comparable to that in real cryo-EM data sets. The number of 20% was choosen because the fraction of contaminations in the data sets is known from own experience to be roughly 20%.

The images for particles and non-particles were generated by projecting 3D volumes evenly distributed into 2D. Making a meaningful statement about the classification performance of our algorithm requires that our model images resembles real cryo-EM pictures in fundamental properties like signal-to-noise ratio (SNR) and image contrast modulated by the contrast transfer function (CTF). This was achieved by the image manipulation procedure described by Frank et al. [Frank et al., 1996].

First, the structural noise in real data sets is simulated by adding random noise with zero-mean Gaussian distribution to a SNR of 1.4. Second, the image formation of a bright field microscope working under $300kV$ and a defocus of 2.0µm was simulated by modulation of the pictures with a contrast transfer function (CTF). The final step was to add random noise (shot and digitization) of zero-mean Gaussian distribution to a SNR of 0.05.

By analogy to image processing of real cryo-EM images, the artificial pictures were also low-pass filtered to reduce the noise. The image manipulation workflow is depicted in Figure 3.16. In order to verify that MAPPOS can cope with all types of contaminations, non-particle images were generated from four 3D templates that served as a projection volume: plate, cylinder, sphere, and void (Figure 3.16-b). These templates were chosen such that they covered the spectrum of contaminations typically encountered in cryo-EM images (Figure 3.16-3c)

The ribosome and contaminant projections were used for the simulation study. The reason is that the classification into particles and non-particles in a real data set is never 100% accurate. Our goal was to compare the performance of human experts with the performance of MAPPOS in a simulation setting which the truth is unambiguously known. The five categories (plate, cylinder, sphere, void and combination of all) cover by far the largest part of all contaminations, thus providing a realistic an representative negative sample set.

It should be noted that in real applications, artificial false projections are not needed, the negative sample set is provided by one or several human experts who pick an initial training set of boxed images with putative particles respectively non-particles.

3.2.2 Validation based on real Cryo-EM Data

In addition to the validation on an artificial data, the performance was also assessed based on a real cryo-EM data set of empty 70*S ribosomes from E.coli*.

Micrographs were automatically collected on an *FEI Titan Krios electron microscope* under low dose conditions. After that the particle picking step was performed and an input data set consisting of $85,726$ windowed projection images which were detected by the template matching algorithm of SIGNATURE was provided.

We compared the performance and the result of the 3DEM process based on three different methods:

- manual post-picking
- no post processing
- post-picking with MAPPOS

Besides their classification performance, we assessed the effect of post-picking on the reconstruction quality of the electron density map.

For automated classification by MAPPOS, a training dataset of $2,000$ particles (50% particles resp. non-particles) was provided. All data sets were processed using SPIDER and refined for 3 rounds to a final resolution of about a Fourier-Shell Correlation (FSC 0.5) of 11Å.

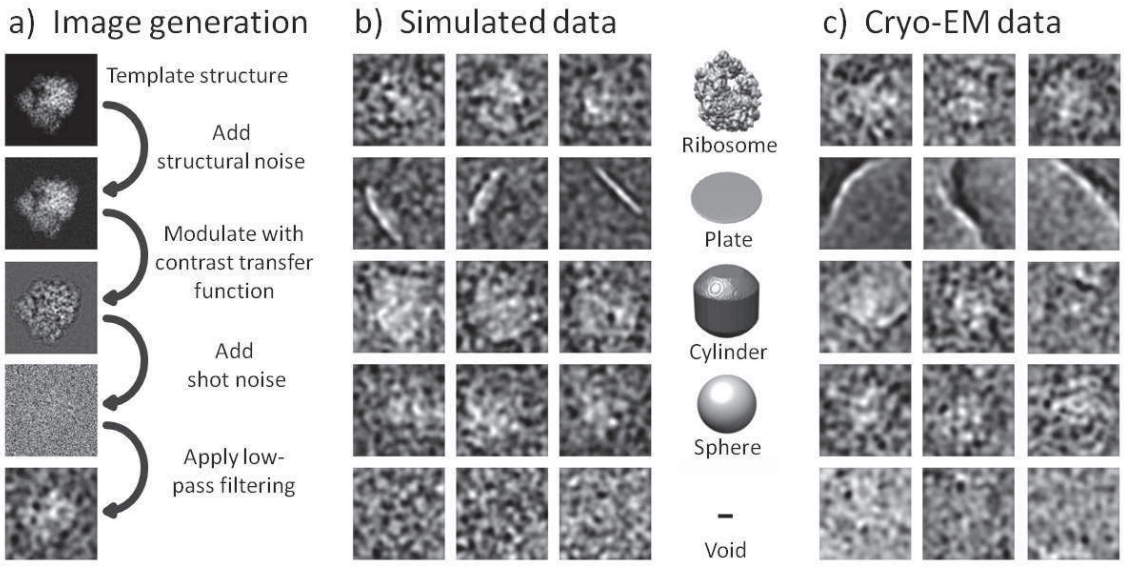


Figure 3.16: Simulation of cryo-EM boxed images.
(a) Generation of an artificial cryo-EM image based on a crystal structure of the *E. coli* 70*S* ribosome (PDB: 2QAL & 2QAM). A 2D projection of the ribosomal electron density is modified by (i) addition of structural noise to account for structural heterogeneity, (ii) distortion with a CTF to simulate the image of a bright field electron microscope at a negative defocus value, (iii) addition of noise to a SNR of 0.05 to simulate low dose conditions, and (iv) low-pass filtering to improve the contrast as routinely done during standard cryo-EM image processing. (b, c) Comparison of experimental cryo-EM images (b) to our artificial projection images (c). Projections from different angles of the E.coli 70S ribosome (row 1) and four types of contaminations commonly found in cryo-EM datasets (rows 2-4) are shown. Various 3D volumes (middle column) were used to generate the artificial (non-)particle images.

4. Experiments and Results

> I have had my results for a long time: but I do not yet know how I am to arrive at them.
>
> Karl Friedrich Gauß(Quoted in A. Arber, The Mind and the Eye, 1954)

This chapter deals with the performance and results of validation routines of MAPPOS. The main part of this chapter is also based on the published paper [Norousi et al., 2013].

4.1 Performance in a Simulated Data Environment

Non-particle contaminations in cryo-EM datasets can severely impair the quality electron density map reconstructions. Taking into account that contemporary automated data collection approaches typically provide an excess of raw data, we focused on a high detection rate for non-particles during the development of MAPPOS at the expense of also removing some particles along the way.

In terms of quantifiable measures we were trying to maximize the specificity [Langlois, 2011] while accepting a lower sensitivity in return (see Table 2.1 for definitions). Each of the five simulation scenarios (see Figure 3.16) was run 100 times, and each run provided a vector (TP, FP, TN, FN) of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN) as its result.

The specificity and sensitivity (as well as their mean and variance values) for each scenario were derived from these values (Table 4.2). Specificity values were above

Contamination type	**TP** ± relative std.dev.	**FP** ± relative std.dev.	**TN** ± relative std.dev.	**FN** ± relative std.dev.	**Sens.** ± std.dev.	**Spec.** ± std.dev.
Plate	814 ± 2%	300 ± 6%	701 ± 2%	186 ± 10%	81% ± 2%	70% ± 2%
Cylinder	865 ± 1%	144 ± 10%	865 ± 2%	201 ± 9%	87% ± 1%	86% ± 1%
Sphere	798 ± 2%	241 ± 8%	759 ± 3%	202 ± 9%	80% ± 2%	76% ± 2%
Void	806 ± 3%	296 ± 7%	704 ± 3%	194 ± 12%	81% ± 2%	70% ± 2%
Mixed	794 ± 2%	257 ± 7%	743 ± 2%	206 ± 9%	79% ± 2%	74% ± 2%

Table 4.2: Performance of MAPPOS in different test scenarios. A set of 1,000 particles and 1,000 non-particles was used for training in each case. For self-containedness, we provide a definition of the performance scores proposed by [Langlois, 2011] for the comparison of particle picking methods. Classification results on the test set were compared to the known labels, splitting the samples into correctly classified particles (true positives, TP), correctly classified non-particles (true negatives, TN), incorrectly classified non-particles (false positives, FP), and incorrectly classified particles (false negatives, FN). Quantities that are derived from these numbers are the sensitivity (=TP/(TP+FN)), specificity (=TN/(TN+FP)), and positive predictive value (PPV=TP/(TP+FP)).

70% in all scenarios, reaching a maximal value of 86% for the cylinder scenario and a value of 74% for the mixed scenario. Sensitivity values were around 80% with the exception of the cylinder scenario with a value of 87%.

4.2 Performance with Simulated Cryo-EM Data

Particles were picked from micrographs containing E. coli 70*S* ribosomes using SIGNATURE and subsequently classified by MAPPOS. The same particles were manually inspected by a human expert whose classification served as a gold standard.

Out of the 85,726 particles, the human expert classified 11,900 as non-particles.

The most relevant quality measure for practical applications is the positive predictive value (PPV), the fraction of particles among all picked images. Note that this quantity is meaningless for simulated data, where the ab initio rates of particles and non-particles, and hence the PPV, can be chosen arbitrarily. The performance on simulated data was measured based on the backprojection of the crystal structure and on variation of training data size.

Based on the backprojection:

We used the backprojection of the crystal structure of the E. coli 70S ribosome and a reconstruction based on the unclassified dataset as a positive and a negative control, respectively. According to *Fourier-Shell correlation* the resolution of the reconstructions was comparable in all cases; however, there were obvious differences in the quality of the density maps. Considering the ribosome, the first features to be resolved are *ribosomal RNA (rRNA)* helices followed by protein α-helices and, later on, β-sheets. One of the evaluated regions was the ribosomal tunnel exit at the ribosomal proteins $L29$ and $L23$ (Figure 4.17). No separation between rRNA and protein densities was observed in the negative control. Accordingly, no information on protein secondary structure information could be obtained.

The reconstructions of the dataset classified by MAPPOS as well as the manually inspected dataset provide information on the localization and secondary structure of proteins. The α-helices of $L29$ and $L23$ are almost completely resolved. Our results illustrate how post-picking of automatically selected particles from cryo-EM micrographs can lead to improved electron density maps, and that post-picking by MAPPOS is on par with manual particle inspection.

Based on variation of the training data size:

We investigated the PPV and sensitivity as a function of the training dataset size (Figure 4.18). Both, sensitivity and PPV increased with the size of the training dataset. The maximal PPV was already achieved with a training dataset of only 1,000 images. The increase of the PPV from 88% (PPV after using SIGNATURE and no further post-picking) to 94% after post-picking by MAPPOS corresponds to a substantial reduction of the fraction of false positives by a factor of about 2.5. Second, as an additional measure of performance, we evaluated the quality of the

electron density maps reconstructed from the MAPPOS or human expert datasets in terms of structural features that were clearly resolved.

4.3 Performance of MAPPOS vs. Human Experts

To investigate the facts how user supervised learning may or may not bias results, how this may be a problem in exacerbating user bias issues and further how are issues of over-fitting avoided (see Section 2.2.4), MAPPOS was compared to 7 human experts using a simulated dataset of $2,048$ images comprising $1,638$ particles and 410 mixed non-particles.

We trained MAPPOS with two different training datasets. The first one comprised 500 true particles and 500 true non-particles, while the second one comprised 500 particles and 500 non-particles that were randomly chosen from the classified particles of the best-performing human expert. The results were analyzed according to sensitivity and specificity.

Notably, as depicted in Figure 4.19, the performances of the human experts were highly variable. The results achieved by MAPPOS were comparable to those of the best-performing human experts. When trained with the first training dataset, MAPPOS scored the 2nd best specificity, achieving 79% specificity and 81% sensitivity. When trained with the random images obtained from the human expert, the specificity (67%) was still above average, while the sensitivity (85%) increased considerably.

We analyzed the individual classified particles according to the types of non-particles that were detected with high or low reliability, respectively. MAPPOS agreed best with the two most accurate (specificity) human experts, and, when MAPPOS was trained by a human expert, it mimicked this experts classification behavior (Figure 4.19 and data not shown).

Pairwise differences between human experts and MAPPOS

Comparison of the classification behavior between human experts and MAPPOS on a data set of 1,638 particles and 410 non-particles (Figure 4.20). We compared of three versions of MAPPOS and seven human experts on the set of non-particles (left) and the set of particles (right).

The color code of each rectangle corresponds to the agreement between two classifiers, 0 indicating no disagreement (100% identical classification behavior) and 1 total disagreement. These agreement values were used to cluster the 10 classifiers by hierarchical clustering (using the function *hclust* in the statistical software R, with average linkage as linkage function and Euclidean distance as distance function).

If MAPPOS has been trained by true particles and non-particles (MAPPOS 1:1, MAPPOS 2:1, see main text), it agrees well with the human experts on both particle and non-particle images. If MAPPOS has been trained by the output of expert 2, it most closely resembles expert 1 and 2 on both types of images; the general agreement on true particles is considerably weaker, due to a generally lower sensitivity.

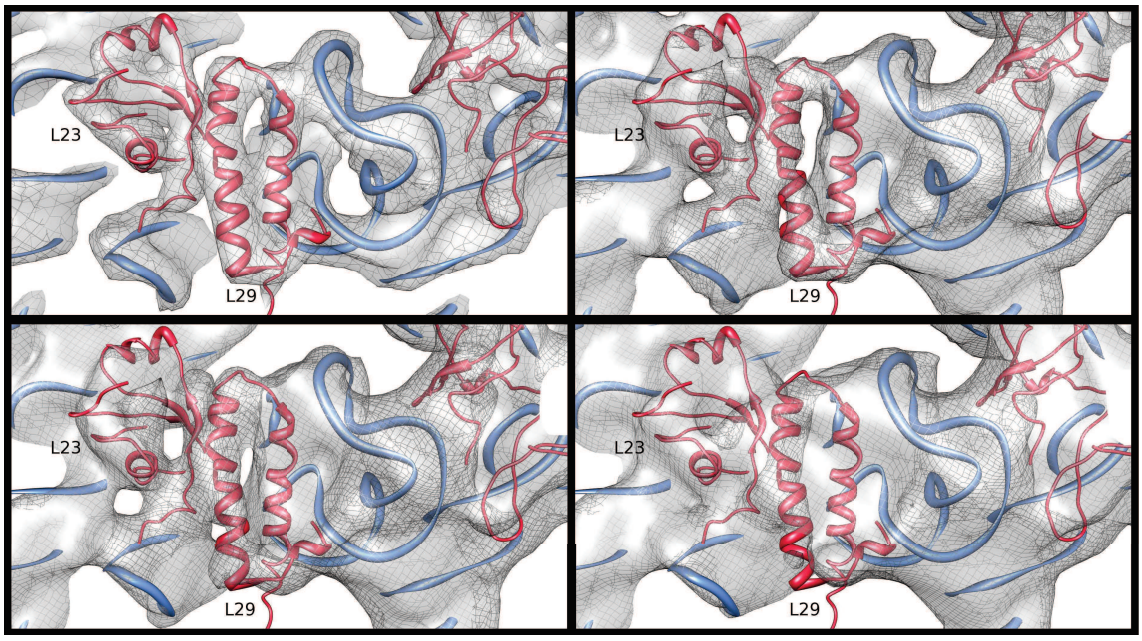


Figure 4.17: Effect of different post-picking classification strategies on cryo-EM reconstructions. A ribbon representation (red: ribosomal proteins; blue: rRNA) of a crystal structure of the E.coli 70*S* ribosome (PDB: 2QAL & 2QAM) was fitted in all electron density maps (grey). The electron density map projected from the crystal structure of the E. coli 70S ribosome and filtered to 10Å resolution serves as a reference (top left). Secondary structure elements, e.g. protein α-helices of $L29$, are resolved in the reconstruction of the manually classified dataset (bottom left) as well as in the reconstruction of the dataset classified by MAPPOS (top right). In contrast, no secondary structure information is resolved in the reconstruction of the unclassified dataset (bottom right).

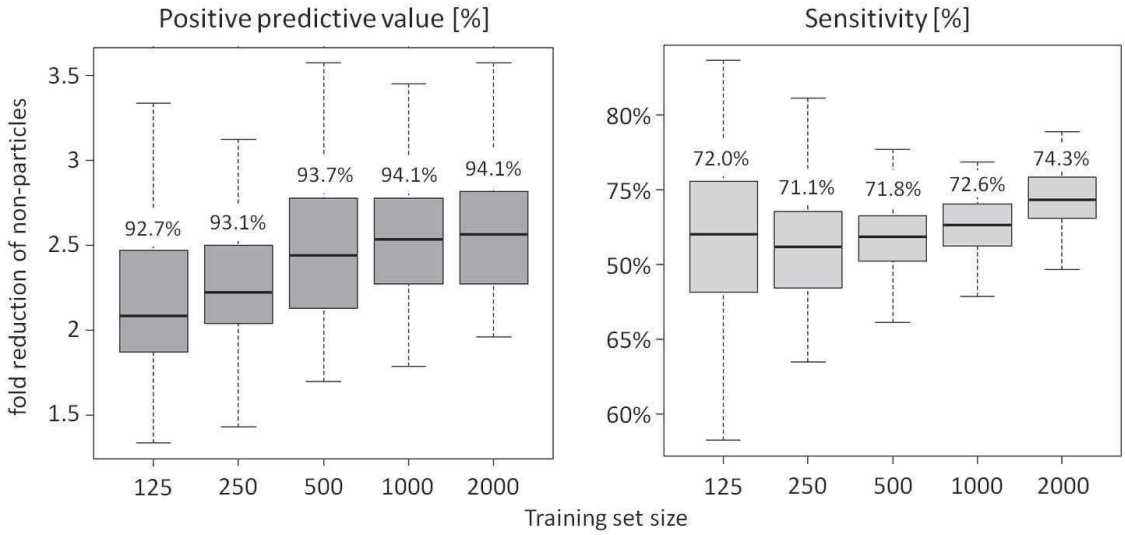


Figure 4.18: Effect of training dataset size on MAPPOS performance.
The PPV (left) and the sensitivity (right) of MAPPOS were tested on experimental cryo-EM data of the E. coli 70*S* ribosome sample using various training dataset sizes (x-axis). Each box summarizes the results of 100 replicate in silico experiments. Each box spans the central range of the data (1st-3rd quartile) while the black lines inside the boxes indicate the respective medians. The whiskers mark the 3-fold inter-quartile range. The fold reduction of the number of non-particles in the dataset (y-axis) is indicated for the PPV.

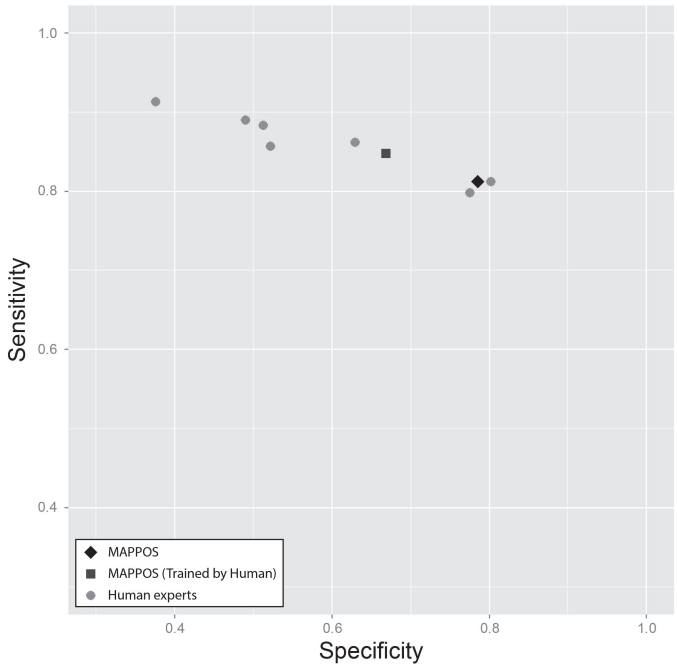


Figure 4.19: Comparison between MAPPOS and human experts.
Sensitivity and specificity values of 7 human experts (circles), MAPPOS trained by the best-performing human expert (square) and MAPPOS trained with true particles and non-particles (rhombus) are shown.

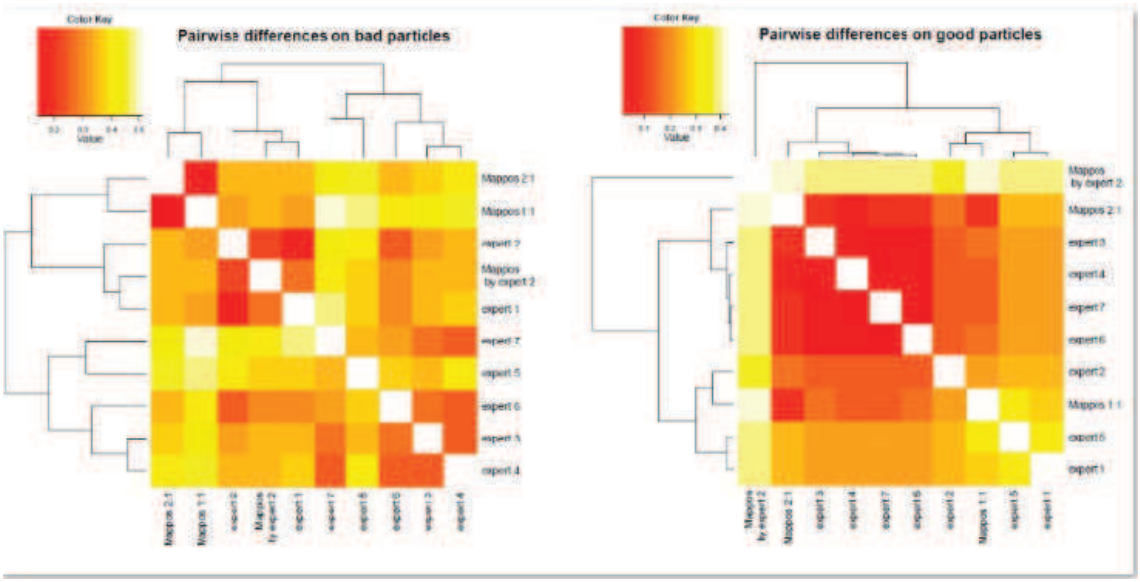


Figure 4.20: Pairwise differences on good and bad particles. Comparison of the classification behavior of three versions of MAPPOS and that of 7 human experts on the set of non-particles (left) and the set of particles (right).

5. Conclusion of Part I

> One picture is worth ten thousand words.
>
> Frederick R. Barnard (Quoted in Printers' Ink, March 1927)

We introduced MAPPOS, an automated image classification method that reduces the manual workload of particle post-picking by orders of magnitude while maintaining a reliable classification quality. We used the example of the E. coli ribosome to demonstrate that the quality of the electron density map after reconstruction from an automatically classified dataset is equal to that of a manually classified dataset.

When compared to human experts, MAPPOS achieved a performance similar to the best-performing human experts. Notably, the performances of the human experts varied considerably (Figure 4.19). An obvious concern resulting from this observation is that MAPPOS cannot perform better than the user that classified the training dataset. However, in the light of large datasets, it is easier for a user to thoroughly assemble a good training dataset than to uniformly judge all of the boxed images with constant high quality. It is exactly this task where MAPPOS outperforms manual classification by impartially applying the same criteria to every image even throughout large datasets.

Existing methods [Nicholson and Glaeser R.M, 2001] for automated particle selection aim at the simultaneous identification and classification of particles on the level of micrographs. The crucial difference in our approach is to address these tasks separately.

We see the major advantage of this strategy in the possibility to provide positive (particle) as well as negative (non-particle) samples that were derived directly

from the dataset itself for the training of the classifier. Such sample images are not available prior to particle picking; only samples from previous datasets or unrelated references can be provided. The use of negative samples that resemble the actual types of non-particles present in a dataset contributes greatly to the specificity achieved by MAPPOS. Consequently, the preceding particle picking step can be highly sensitive (filter criteria can be less strict), because a sufficient specificity is ensured by the subsequent post-picking step.

In this study, we demonstrated the applicability of MAPPOS to experimental cryo-EM data on the example of an E. coli $70S$ ribosome dataset. While the application of MAPPOS is probably most beneficial for unsymmetrical specimen since large numbers of particles are required for cryo-EM reconstructions in such cases, it remains to be shown that it is applicable to a broader range of particles.

Notably, MAPPOS has already been successfully used for high-resolution reconstructions of $80S$ ribosomal complexes ([Becker et al., 2012], [Leidig et al., 2012]).

Current generation electron microscopes (e.g. *FEI Titan Krios*) generate up to $4,000$ micrographs per day using automated data acquisition techniques. The Titan Krios is a 300 KV transmission electron microscope ideal for high resolution cryo-electron microscopy or cryoelectron tomography. For ribosomal samples this amounts to roughly $200,000 - 500,000$ particles per day. Automated particle picking and post-picking tools are therefore likely to become an integral part of cryo-EM processing pipelines. Manual classification of the E. coli $70S$ ribosome dataset used in this study required 3-4 working days although it comprised merely $85,726$ boxed images.

In contrast, the classification using MAPPOS required less than a day including the manual generation of the training dataset. Despite its high speed, the quality of the final 3D reconstruction was equivalent to that of the manually classified dataset. This demonstrates that MAPPOS is able to handle huge amounts of data at the same pace at which they are generated without impairing the quality of the resulting electron density maps.

Part II

Spot Detection and Colocalization Analysis in 3D Multichannel Fluorescent Images based on Spatial Statistics

6. Introduction

> Every great advance in science has issued from a new audacity of imagination.
>
> John Dewey
> - An American philosopher and educational reformer-

It it is widely recognized that subcellular objects (e.g. proteins) fulfill some essential cell functions. This takes place by interacting with each other in a highly regulated fashion [Helmuth et al., 2010]. In recent years, well-developed cell imaging techniques have considerably improved our understanding of cell structures and functions. In order to understand how diverse biological processes have been carried out, it is essential to establish accurate approches for detecting and localizing of genes [Helmuth et al., 2010].

This chapter introduces the most commonly used image acquisition and labeling methods of *Fluorescence microscopy* and describes approaches which are able to analyze the colocalization quantitatively. Furthermore, the main challenges in this field are described extensively. Finally an overview of our contribution during this work concludes this chapter.

6.1 Fluorescence Microscopy

In order to gain insight into the world of cellular biology, the method of *labeling* is a common used way. It facilitates an analytical investigation for biological purposes like molecular interactions or protein localizing. Currently, most of the labeled biological image data is collected using *Fluorescence Microscopy* and the acquired images are subsequently analyzed quantitatively by an appropriate software [Ronneberger et al., 2008].

Fluorescence is based on the process that whenever light comes in contact with a molecule of a specific wavelength, first the light will be absorbed and the molecule will emit the light of longer wavelengths (i.e. of a different color than the absorbed light). In more details, at initial condition most molecules are in lowest energy state, but after absorbing a photon their energy causes the electron to jump to an excited state [Murphy, 2001]. In other words, the component of interest in the specimen is selectively dyed (selected) with a concentration (fluorescent molecule) called *fluorophores* or *fluorescent dyes* [Gonçalves, 2009].

The fluorescence microscopy has a high performance and permits the observation of subcellular events which are not always feasible by conventional methods. It provides labeling subcellular structures with a high sensitivity (degree of correctly detecting components) of complex biomolecular assemblies [Ronneberger et al., 2008]. A further advantage of fluorescence microscopy is that it allows a multicolor visualization of cellular components in a suitable resolution [Murphy, 2001].

6.2 Colocalization Analysis

Colocalization is a common used method that can be considered as a multifaceted concept in cellular biology. It is therefore multifaced because it can be analyzed as well as in a physical, in a biologically and in a imaging context. For instance, in a biological context, it means that two different molecules attach to the same structure within the cell to fulfill a biological function. In the imaging context it is described as the spatial overlap of two or more dyes in a multichannel image that can be interpreted as a colocalization in this location [Maanders et al., 1993].

In general during the colocalization analysis, two proteins (provided in two channels) are stained with a fluorescent dye or fluorophore and they are subsequently imaged using an appropriate microscope. Thereby some snapshots over the time are taken to measure the intensity and locations of their subcellular objects. To investigate the colocalization degree there are two possibilities, it can be either performed *optically* or *quantitatively*.

Optically: This type of colocalization analysis is called also manual or qualitative analysis. Normally one of the channels is colored green and the other one red and in the next step these two images are overlapped in order to evaluate the colocalization intensity in an optical manner. As shown in Figure 6.21, those locations where the subcellular of both channels have the same coordinates, lead to a yellow appearance in the pixels. According to the yellow intensity of overlapped pixel, the strength of the interaction can be recognized. This kind of analysis is called qualitative and is rather a subjective task since the results of different analyses are incomparable and it is difficult to find a suitable threshold about the colocalization term.

Quantitatively: Due to the drawbacks when using optical colocalization analysis, the colocalization analysis based on quantitative approaches is preferred which performs statistical analysis and quantitative methods in order to achieve automatically suitable colocalization results. The goal is to achieve results which are as objective and as accurate as possible. These techniques are describes in 7.3.

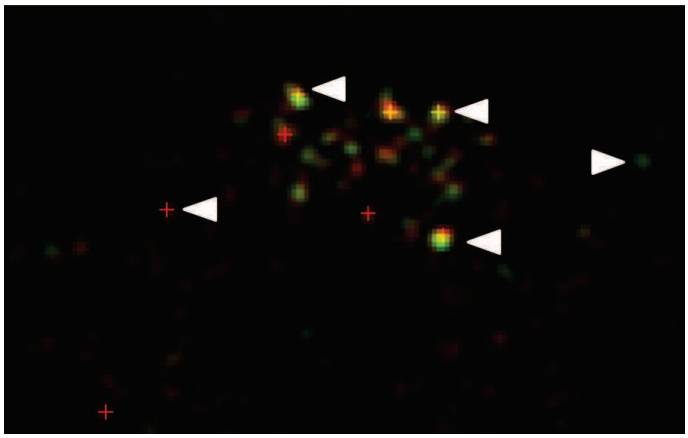


Figure 6.21: Sample of optical colocalization. Two channels green and red are merged (overlapped) after object detecting. The yellow points show a colocalization and other points which are either red or green indicate no-colocalization.

6.3 Challenges of Colocalization in Fluorescence Microscopy

Initial investigations reveal that the colocalization analysis is a very challenging task due to the heterogeneity of acquired images regarding their pixel intensity, point distribution and illumination. Analyzing Fluorescence images shows that the density and the distribution of objects in different regions differ considerably. Furthermore the regions are distinguished by *Signal-to-Noise ratio (SNR)*. Therefore the object detection step should be performed taking into account these varieties and appropriate techniques should be applied depending on the properties of the various objects. Furthermore, it should be also noted that biological structures are distorted along the z-axis due to the discrepancy between lateral and axial resolution of optical microscopes.

A further challenge is at the *X-chromosomal space* of the nucleus. As depicted in Figure 6.22, the molecules come into close contact (there is a high density). Thus, the noise of single molecules come very close and eventually overlap with each other and lead to a so-called „big objects“. In this case the goal is to detect those subcellular objects separately which together form the big object. The effect of close-by objects and should be considered during the image acquisition described in Section 7.1.4 and Figure 7.28.

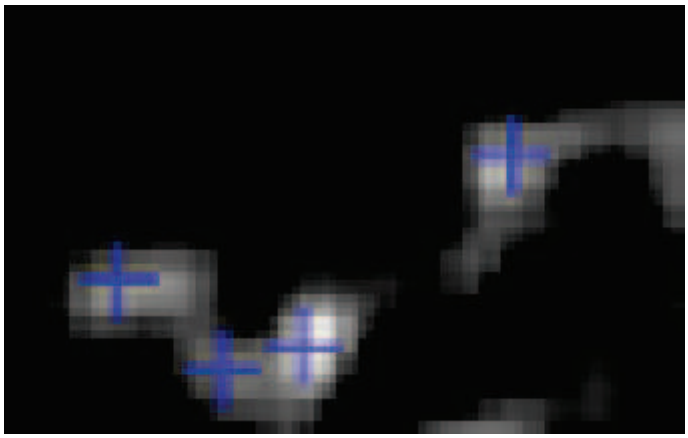


Figure 6.22: A cropped region of MSL2 molecule. It illustrates a challenging case when two subcellular objects are very closed-by and their regions are overlapped which should be detected as two separate objects.

Moreover from a biological point of view, the chromosomes are not randomly organized during interphase but instead they form a discrete cluster called *chromosome territories (CT)* . The field of nuclear topology is in transition from a pure qualitative and descriptive field towards a more quantitative field boosted by advances in super resolution microscopes which allow visualization of detailed properties of chromosome architecture along with regulatory protein complex. Still, the great challenge is the quantitative evaluation of images, in particular towards describing complex protein staining patterns [Schermelleh et al., 2010].

We assume that objective assessment on foci number and distribution promises to obtain quantitative information on the distribution of protein foci accumulations on a single cell. Combined with quantitative information from orthogonal population-based methods (biochemical counting of molecules) and genomics allow to derive quantitative models to predict the function of (multi)-protein complexes on structural organization of CTs.

The associated key questions for our investigation can be formulated as follow:

- How many proteins of a given species per cell can be counted?
- At which loci[2] does the protein localize in a cell population average?
- How many foci[3] appear in a region?
- Which volume does it occupy?
- Is a fluorescent signal distributed on a membrane surface or is it contained throughout the cytoplasm as a soluble factor?
- Are different fluorescence signals colocalized on the same structure within the limits of resolution of the light microscope?

Another challenge in this field is the fact that the images must be analyzed fully in 3D in order to obtain the entire information about the subcelullar objects. Some foci are ambiguous whether they reflect biological meaningful signals or technical noise. Therefore appropriate strategies have to be developed to cope with technology complications (signal-noise, adjacent objects). The high number of objects in 3D which strongly impairs manual assessment, and robustness towards technical and biological variation.

More problematic is the case where the microscopes' acquisition and processing methods are uneven. On the basis of our internal studies and analyses we noticed that frequently the dynamic range and other essential image properties of two channels of the same object are distinct. This leads to misinterpretations and difficulties in colocalization analysis. Inaccurate interpretations regarding colocalization in merged color images is caused by several factors, including improper gain and offset settings of camera during acquisition or from extreme histogram stretching during image processing.

[2]Loci is plural of locus, which means in genetics sense the location of a gene (or of a significant sequence) on a chromosome, as in genetic locus.

[3]Plural of focus. The origin or centre of a disseminated disease.

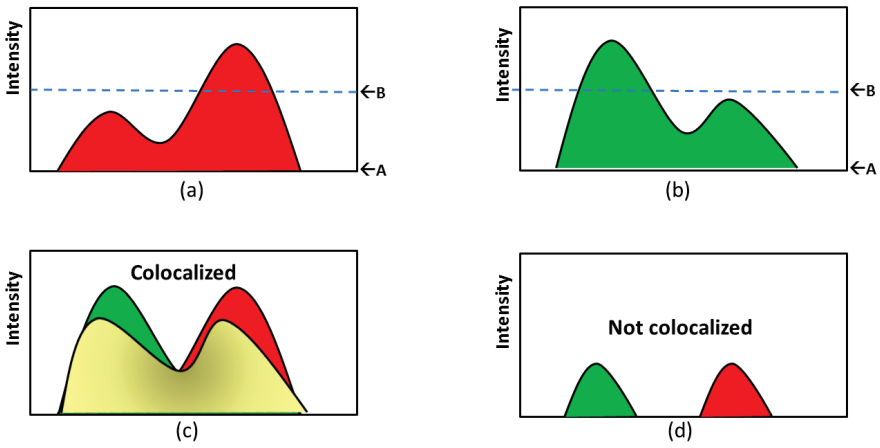


Figure 6.23: Colocalization of two fluorescent signals based on determining the offset. (a-b) show a red and green fluorescent signal intensity versus spatial distribution. In each diagram the offset threshold is set on intensity value A (solid line) and B (dotted line), respectively. These two different values lead to 2 different colocalization results. (c) is interpreted to be partially colocalized and (d) having distinct distributions. One gets different results depending on the offset applied on the image.

6.4 Our Contribution

Since manual object detection is very inaccurate and time consuming, some automatic object detection tools have been developed in recent years. At the moment, there is no available image analysis software which provides an automatic, objective assessment of 3D foci which is generally applicable and freely available. Complications arise from discrete foci which are very close or even come in contact to other foci, moreover they are of variable sizes and show variable signal-to-noise, and must be analyzed fully in 3D.

Some of well-developed tools are commercial like the *Imaris* (developed by Bitplane[Andor, 2013]), further the *Volocity Image Analysis Software* (provided by PerkinElmer Group) and the *Cell Profiler* [Carpenter et al., 2006]. They all provide the necessary functionality for data visualization, analysis, segmentation and interpretation in 3D or 4D microscopy data set with a very high speed. But their main disadvantages are the lack of flexibility and sufficiently user participation in the analysis process. I.e., they can just be adjusted by a few parameters in advance and the user can not follow up the sub steps during the analysis. Hence it induces to a

kind of having „black box“ which is able to analyze specific types of image data.

Other open-source tools like the *JACoP*[Fabrice P. Cordelieres, 2006] developed by [Bolte and Cordelieres, 2006] and the 3D Object counter [Rasband, 1997] both as a plugin of *ImageJ* software, offer users more opprotunities to look into the code, but the functionalities and abilities of these tools are very restricted and the tools are very problem-specific.

Therefore an automatic analysis toolbox with the option of manual intervention in order to get a more accurate and flexible analysis is of great importance. This is the main objective of the project and the realized results show that the chosen and proved approach have been reasonable and professionally applied to deal with the mentioned requirements.

We introduced *3D-OSCOS* (**3D**-**O**bject **S**egmentation and **Co**localization Analysis based on **S**patial statistics) a more general multistage segmentation approach, which considers the heterogeneous of the desired fluorescently labeled spots regarding their size, intensity and pairwise distances. 3D-OSCOS is a fully automatic yet still manual re-adjustable and flexible pipeline for spot detection from nuclear compartments.

The algorithm is divided into two stages: spot detection and colocalization analysis. Spot detection consists of 3D image smoothing and filtering to reduce the effect of noise, intensity threshold, followed by a multistage segmentation approach for object detection, concluded by measuring the properties of detected objects (e.g. coordinate, size, etc.). The second step analyses and measures the inter-distances between centers of objects in order to ascertain a statement about the colocalization degree. The first step is developed in MATLAB and the second part is developed in R. Both developed tools are freely available in the community for research applications.

The 3D-OSCOS gives the users the ability to drive their experiments analysis in either automatic or semi-automatic detection mode. Farther in this work we established a connection between colocalization and *spatial point process* in order to investigate whether an observed result of colocalization could have occured by chance or not. Spatial point process exploits the information contained from the data more extensively than the classical colocalization analysis by determining the statistical significance. The spatial point process deals with analyzing the geometrical structure

of point patterns which are derived from the location of detected subcellular objects.

We also verified our method with both a real experimental and an artificial (computer-generated) data set in order to evaluate the performance against a known ground truth.

In order to get an insight into one of our investigated 3D image data set and the goal of coloclization analysis, Figure 6.24 shows all 45 slices by overlapping the red and green channel. In each slice three different object colors (red, green and yelow) can be seen, where yellow color indicates an overlapping effect of the same object from the green channel on the red channel. This is the typical approach for visual inspection of colocalization analysis.

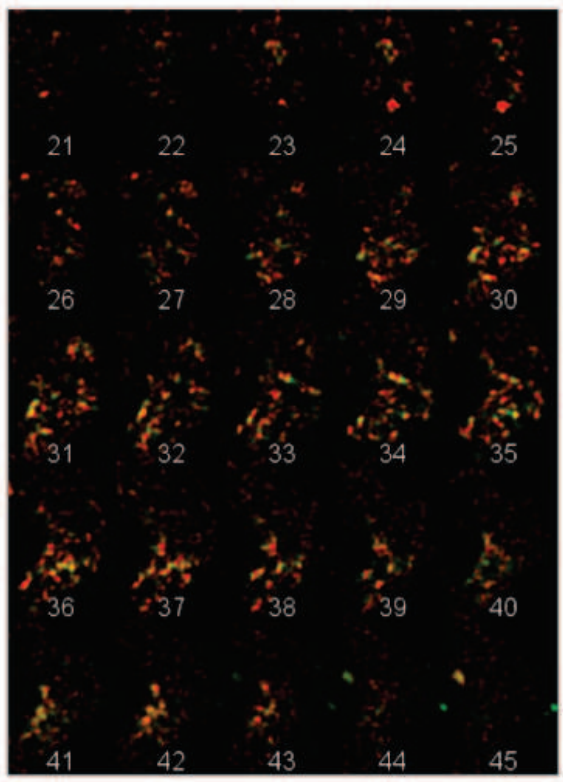


Figure 6.24: An illustration of MSL protein binding in two channels red and green over all slices. The yellow regions indicate a colocalization between objects from both channels.

7. Theory Basics

> Most of the fundamental ideas of science are essentially simple, and may, as a rule, be expressed in a language comprehensible to everyone.
>
> Albert Einstein

This chapter covers the theoretical background of image acquisition techniques. Further, some common spot detection and colocalization approaches in fluorescence images are described. Finally the idea of interpreting the observed subcellular objects as a realisation of a spatial point process, is introduced. First of all, the theory and practical issues of imaging and quantitative analysis are introduced, which are essential in order to comprehend the process better.

7.1 Image Acquisition Techniques

It is well recognized that imaging and visualization have an essential role in cell biology. Therefore many researches in modern biology are based on looking into a biological object. Already in the late seventeenth century, microscopes enabled many valuable insights of a minute world that was not possible to see with the naked eye (Robert Hooke and Antoni van Leeuwenhoek[4]). Since that time microscopes have played an important role in many aspects of our society, particularly in sciences.

Microscopes are used to magnify very small objects and to bring them into a form which is convenient for analysis. They are roughly based either on light or electron signals where the different technologies vary in terms of resolution, specifity, complexity and in price. For better understanding the most important and

[4]Micrographia Some Physiological Descriptions of Minute Bodies Made by Magnifying Glasses with Observations and Inquiries Thereupon.

commonly used imaging microscopes are introduced in next sections. In addition to the Electron Microscopy (EM) which is introduced in 1.1, further techniques such Light Microscopy, Confocal Laser Scanning Microscopy (CLSM) and 3D Structured Illumination Microscopy (3D-SIM) are described in next sections.

7.1.1 Light Microscopy

Light microscope (or optical microscope) is the most valuable and important tool in the cell biology to understand the structure and function of cells, which was historically used as a qualitative technique. But around 30 years ago, Belmont et al. [Belmont et al., 1986] identified the need for methods of both quantitatively acquiring and analyzing data in a statistically meaningful way. The light microscope consists of a system of lenses and uses visible light (photon) in order to capture the signal from objects. It allows to localize cellular components up to a certain limit of object size by measuring an inteaction between the photons and an objects [Murphy, 2001].

The main drawback of light microscopes is their limited resolution, due to a series of fundamental physical factors of the light wavelengths. The low resolution of conventional light microscopes comparative to the scale of subcellular objects, prevents researchers from seeing more details within cells [Schermelleh et al., 2010].

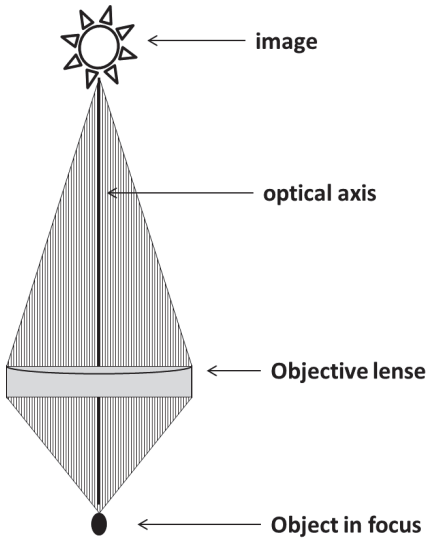


Figure 7.25: An illustration of a simple light microspcope.

7.1.2 Confocal Laser Scanning Microscopy (CLSM)

As mentioned before, conventional light microscopes create images of specimen which are blurred and are not appropriated for subcellular objects. Due to the limitations of conventional light microscopes, improving the resolution while keeping their advantages has been a longtime goal in recent years. Therefore, the *Confocal Laser Scanning Microscopy (CLSM)* was introduced. It is able to exclude most of the light from the specimen which is not from the microscope's focal plane, which leads to create sharp images [Sheppard and Shotton, 1997].

The idea of confocal microscopy was first pioneered by Marvin Minsky in 1995 [Minsky, 1988], where a specimen is captured by focusing a point of light sequentially across it and collecting this information to a unique structure.

As illustrated in Figure 7.26, it has an additional lens, so called *confocal pinhole*, which allows only light from the plane of focus to reach the detector and eliminate out-of-focus light in specimens by suitably positioning of the pinhole. This helps us to get a reasonable resolution and clarity. Further it allows user to collect the image information in form of slices for use in creating a 3D reconstruction of the cellular object.

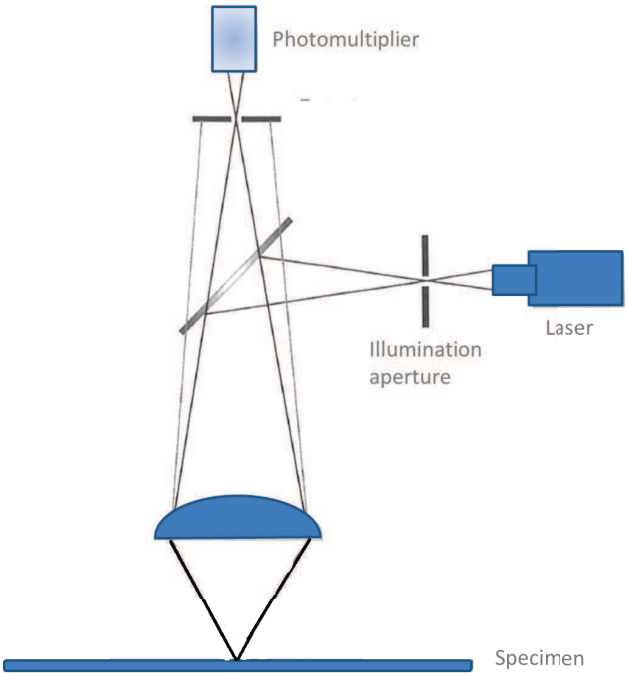


Figure 7.26: Confocal microscope with ray diagram. Ray diagram of confocal optical arrangement, showing how light rays originating away from the plane of focus are eliminated from the image by the detector aperture.

To sum up, it can be said, that the images acquired by confocal microscopy have higher resolution (especially in the z direction) and much less out-of-focus-blur which makes it very convenient for image acquisition. These make images crisper and clearer with more details (see Figure 7.27). In order to avoid a noisy image, each point should be illuminated for a long time to collect enough light to make an accurate measurement [Wright and Wright, 2002]. The drawback of confocal microscopes is that it is restricted to visualize only small volumes at once.

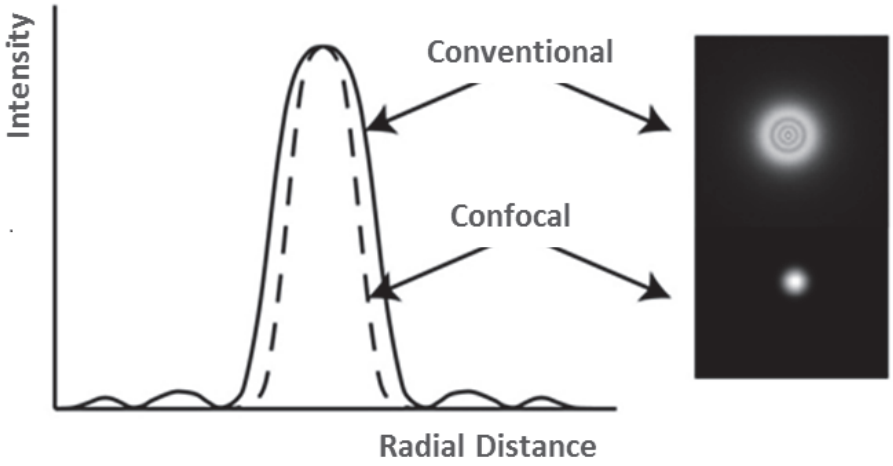


Figure 7.27: Intensity profile of a point source of light. It is illustrated that how a confocal microscope removes the out of focus light, hence the yield intensity profile is sharper than when using a conventional microscope. This leads to the fact that the resolution of the confocal microscopes are much higher than conventional microscopes.

7.1.3 Structured Illumination Microscopy (3D-SIM)

Structured Illumination Microscopy (3D-SIM) is a microscope system based on three-dimensional structured illumination systems. This approach breaks the diffraction limit of light by protecting structured light patterns onto the sample and measuring the frings in the *Moire pattern* form the interface of the illumination pattern and the sample [Gustafsson, 2005].

The illumination pattern interacts with the fluorescent probes in the sample to generate interference patterns. It can increase the spatial resolution of conventional fluorescence microscopy beyond its classical limit by using spatially structured illumination light [Schermelleh et al., 2008].
Furthermore it allows a visualization of cellular objects in multicolor with a high specifity.

7.1.4 Principles of Digital Imaging

Fundamentals of Digital Imaging

It is very important to emphasize that sampling and image acquisition are very crucial steps because some of the structural damages or suboptimal quality of acquired images cannot be enhanced and repaired by subsequent steps even using perfect image processing tools. Therefore understanding the acquisition parameters, their impact on the quality and choosing suitable values for them are of prime importance.

In case of quantitative purposes, a so-called *digital imaging system* is required. In general, the digital imaging system is a combination of a microscope, a digital camera and an imaging software. In general, the job of microscopes is to collect as much as possible of the emitted light or fluorescence given off by the object in order to facilitate visualization of fine detail [Murphy, 2001].

The quality of the acquired image depends on some crucial factors that can be roughly divided into *optical properties* of the microscope (e.g. objective lens) and *acquisition settings* (e.g. resolution, dynamic range etc.). Referring to [Murphy, 2001] and [North, 2006], some of the most relevant parameters are described as follow:

Magnification: As mentioned before, magnification is the effect of apparent enlargement of an object by an optical device. In other words, the magnification is just used to increase the apparent size of objects until they can be perceived by human eyes.

Numerical Aperture (NA): The Numerical Aperture (NA) describes the intensity of the signal captured by the microscope. It is defined as the sine of the maximum angle from the optical axis at which light can enter. Considering the properties of a lens, it is important to distinguish between the well-known magnification of an objective and the few-known NA that describes the light gathering ability. Greater value of NA means more signal is emitted and hence more details are captured. It is important to note, that not the magnification but rather the numerical aperture is the most important in resolution [North, 2006].

After setting appropriate microscope parameters, further image acquisition parameters should be set. According to suitable acquisition settings in order to obtain meaningful and quantifiable images. First it should be noted that most of the images are acquired for additional quantitative analysis by extracting meaningful

information from them. In general, the quality of the acquired image which is associated among other with some essential imaging properties. Following definitions from digital imaging approches (all quoted from [Murphy, 2001]) are essential for understanding the most important fundamentals of digital image processing:

Resolution: The spatial resolution is defined as the smallest distance between two small objects that can be still discerned as two separate objects. The resolution is a much more important factor than the magnification (defines the degree of enlargement of an object provided by the microscope).

The resolution, as shown in Figure 7.28, describes the ability to recognize and distinguish two closely-spaced objects as being separate entities. Hence if these objects are not captured it is related to the fact that the distance of them is out of the resolution range of the microscope (unresolved). Therefore the smaller the resolution, the better we see details of the image.

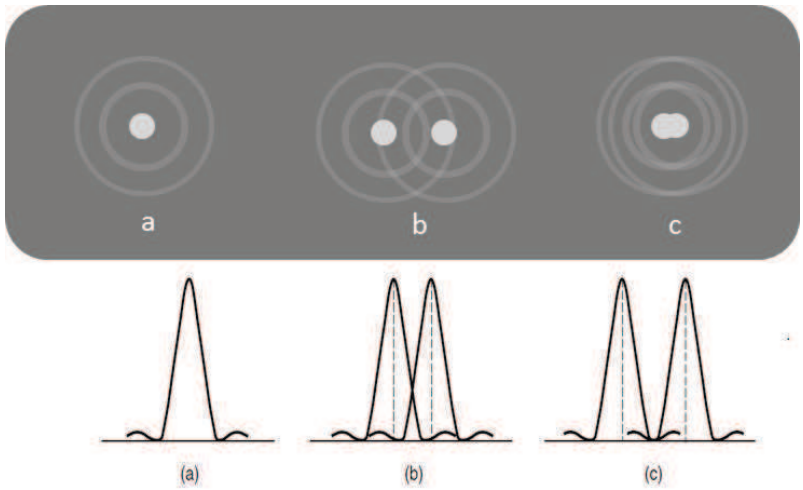


Figure 7.28: The effect of resolution on closely-spaced objects. The effect of resolved and unresolved objects are illustrated in samples c) and b), respectively. In the right image two white dots are seen as separate objects (they are said to be „resolved“) due to the high resolution. Image b) however shows only one large diffuse dot, even though it actually represents the same two separate objects seen.

Signal-to-noise ratio (SNR): As it is well-known, in order to image a cellular object using a microscope, an amount of photons must be detected at the detector. This detected signal is then amplified and displayed as a pixel intensity value in the image. Thereby the *SNR* is calculated as the specimen signal divided by the noise of the surrounding background. According to Sheppard's definition [Sheppard and Shotton, 1997], SNR is the ratio of the „useful" fluorescence signal and the variation (which may be determined as the standard deviation) in that signal. Thus the goal is to maximize the quantity of SNR as much as possible under considering of the imaging time. In quantitative terms, SNR is used to describe the clarity and visibility of objects in an image.

Noise: Noise is considered here to be the random variation within images. Noise originates from a variety of sources including inherent, random, Poisson distributed statistical noise associated with photons. Noise plays a major role in the determination of image quality as defined in SNR.

Dynamic Range: The dynamic range (DR) is the number of resolvable steps of light intensity, described as gray-level steps ranging from black to white which are indicated in *bit depth*. Dynamic range is used to described the potential number of gray-level steps capable of being recorded by a camera, which described . Since a computer bit can only be in one of two possible states (0 or 1), the bit depth is described as 2^x number of steps. Therefore, 8-,16- or 32-bit processors can encode 2^8, 2^{16} and 2^{32} gray levels. For puposes of accurate quantification of light intensities, it is recommended to use a larger number of gray-levels (e.g. 32-bit).

Sampling and Oversampling: Sampling deals with determining and setting the adequate voxel size for image acquisition. This is the next important aspect that ensures the appropriate x-,y- and z- dimensions which in order to recover correctly all information of the cellular object.

These dimensions are determined according to the *Nyquist sampling theorem*, that says, that the voxel size must be 2.3 times smaller than the smallest resolvable feature size of the specimen. Another way to say this is that our voxel size needs to be a minimum of approximately 0.4× the dimensions of that object. For example, if the theoretical resolution limit in the lateral (x,y) plane is 0.3µm, then we must use a pixel dimension of $0.3\mu m \cdot 0.4 = 0.12\mu m$ (or 120nm) in that plane.

Image Processing Methodologies:

Common image processing steps can be roughly divided into the *high-level image processing* (e.g. image analysis or segmentation) which is more application-specific, and the more general approach *low-level image processing* (e.g. histogram adjustment). Low-level processing denotes manual or automatic techniques, which can be realized without a-priori knowledge on a specific content of the image [Gonzalez and Woods, 2008].

Image Processing with Filters: Filtering is used to sharpen or blur an image by *convolution* which is an operation that uses the weighted intensity of neighboring pixels in the original image to compute new pixel values in a new filtered image. A matrix kernel of numbers, called *convolution matrix*, is multiplied against each pixel covered by the kernel, the products are summed, and the resulting pixel value placed in a new image. Only original pixel values are used to compute the new pixel values in the processed image. The *kernel* or *mask* can have different sizes and cover a variable number of pixels such as 3×3, 5×5, 7×7 and so forth. Note that the sum of the numbers in the kernel always add up to 1, as shown in Figure 7.29.

Filter operations can be initially categorized into *Low-pass filter for blurring* and *High-pass filter for sharpening*. The lowpass filter removes high spatial frequency, such as sharply defined intensity differences at the edge of objects in the image. Whereas highpass filtering differentially emphasizes fine details in an image and is an effective way to sharpen soft, low-contrast features in an image. Both categories belong to the *linear filters* which replace the intensity value of each pixel by weighted average over the pixel values of its neighborhood. In general filtering is used to decrease the effect of acquisition errors, suppress the imaging noise and enhance the signal [Gonzalez and Woods, 2008].

$\frac{1}{25}$ x

1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1
1	1	1	1	1	1	1

$\frac{1}{139}$ x

1	1	2	2	2	1	1
1	2	2	4	2	2	1
2	2	4	8	4	2	2
2	4	8	16	8	4	2
2	2	4	8	4	2	2
1	2	2	4	2	2	1
1	1	2	2	2	1	1

Figure 7.29: Two samples of 2D low-pass filters with 7×7 kernels. Both these filters serve toward smoothing images. Left a averaging and right a Gaussian filter where its weights are samples of Gaussian function with $\sigma = 1.4$

7.2 Spot Detection and Quantification in Fluorescent Images

The detection of spots (subcellular objects) and their quantification in multichannel 3D fluorescent images are generally accepted as key steps in order to understand how the spatial organization is established. Further they help us to gain local information of proteins distributed in nuclei. This information is crucial to describe the role of spot locations in biological processes [Helmuth et al., 2010].

In general, image acquisition of small subcellular objects pursues two main objectives. The first objective is to provide an insight into phenotypes and cellular functions [Pepperkok and Ellenberg, 2006]. The other objective is to detect usually small objects with the highest reliability in the presence of other often unknown substances [Trabesinger et al., 2001].

In the following sections current object detection and colocalization methods are introduced.

7.2.1 Manual Detection and Quantification

Manual detection and quantification of spots in 3D images is very laborious and complex, due to the spots spanning across multiple slices in a 3D space. It is also a very time-consuming task, because of the increased imaging throughout and the

large-scale data acquisition of current generation microscopes [Xiaobo Zhou and Wong, 2006].

Furthermore the manual quantification is user-dependent and inaccurate as it is a subjective task. I.e., different experts will have different outcomes. Therefore the analysis of cellular structures based on 3D images obtained with fluorescence and confocal microscopes requires accurate and automatic detection [Ruusuvuori et al., 2010].

7.2.2 Automatic Detection and Quantification Methods

In order to prevent the laborious manual quantification, various automatic 2D and 3D approaches have been developed. First generation of automatic detection and quantification of subcellular objects was based on 2D approaches (slice-by-slice). Summarizing these methods, it can be said that the object segmentation and quantification step is performed among others by threshold-based methods [Fay, 1997], edge detection-based techniques [Jaskolski et al., 2005], watershed transformation [Woolford and Hankamer G. Ericksson, 2007] and 2D Gaussian fitting [Trabesinger et al., 2001], which are performed in 2D (slice-based) without taking into consideration the expansion of the objects in 3D image.

Analyzing of colocalization between different 3D image channels based on 2D slice-based technique leads to inaccuracies and underestimates the number of colocalzations, since two colocalized objects could be visible in 2 adjacent slices but they could not be detected in one slice. Therefore a reasonable detection and quantification method should carry out the full information and reflect the nature of biological samples (e.g., proteins) in three dimensional spaces.

Moreover, due to the discrepancy between lateral and axial resolution of the confocal microscopy (e.g., 4 µm in x-,y- and 12 µm in z-direction, see Section 7.1.2), spots are usually distorted along the z-axis. Thus, for an accurate analysis, the degree of distortion by the optical device should be considered. Otherwise, analyzing colocalization events in two-dimensional images leads to misinterpretation and incomplete spatial description due to missing information about the size in z-direction.

Facing these problems, several (semi-)automatic quantification and colocalization methods have been developed, most of those are very problem-specific. Their great efforts is the possibility to analyze spots by exploiting the full information of 3D

structure data. Among others the following commonly used methods are described in more detail:

Worz et al. [Worz et al., 2010] introduces an automatic 3D geometry-based quantification of colocalization between two channels using three dimensional geometry structures of objects. The approach consists of two main steps. The first step is *3D spot detection step*, where different 3D image filtering and smoothing operations are applied, in order to obtain coarse spots and to reduce the noise. The image is convolved by a 3D-Gaussian filter with a standard deviation σ_f proportional to the desired spot width. The second step is the *spot quantification*, where each of the previously detected spot candidate is evaluated and fitted to a 3D-Gaussian function using least-squares fitting model to specify the properties (e.g. size, structure, coordinates etc.). In the image data the 3D intensity profile around a defined voxel (seed point specified using local maximum search) can be well presented by a 3D Gaussian function $g(x)$. Among different 3D Gaussian models g_M with various parameters, the model with the lowest value for the objective function:

$$\sum_{x \in ROI} (g_M - g(x))^2$$

is chosen as the best fitted Gaussian function. The parameters of the best selected function specifies and defines the properties of the region.
The computation time is relatively short (approximately 3s - 5s) with a good performance. Unfortunately this tool is not freely available.

Ram et al. [Ram et al., 2010] developed a method to segment and classify 3D spots in fluorescence images. They applied this approach on two application fields, first on in-situ hybridization images from *ovarian germline nurse cells of Drosophila melanogaster* and on *3D FISH images* to detect and localize the presence of specific DNA sequences.
The algorithm mainly consists of two steps *spot segmentation* and *spot detection*. The spot segmentation consists of 3D smoothing, top-hat filtering, intensity thresholding and 3D region growing. The spot detection is based on machine learning approaches. After spot segmentation a number of discriminative features (e.g. volume, contrast, texture etc.) are extracted from them and based on these features a classifier is generated to classify the spots as either true or false spots (spot detection phase).

Raimondo et al. [Raimondo et al., 2005] proposed a multistage algorithm for automated classification of FISH images from *breast carcinoma samples*. The algorithm consists of two stages: nucleus segmentation and spot segmentation using different techniques. The nucleus segmentation step consists of a nonlinear correction, global thresholding and marker-controlled watershed transformation. The spot segmentation step consists of top-hat filtering, followed by thresholding and gray-level template matching to separate real signals from noise.

Bolte et al. [Bolte and Cordelieres, 2006] introduced an object-based analysis to detect and segment spots in FISH images. Furthermore they provide an online available IMAGEJ tool called three-dimensional object counter ([Fabrice P. Cordelieres, 2006]) that uses the object-based colocalization analysis and allow an automated colocalization analysis in a three-dimensional space.
In the segmentation phase the foreground regions are emphasized using a 3D anisotropic smoothing to remove the effect of noise, 3D-top hat filter to represent the foreground details better, binary thresholding and 3D region growing. In the classification phase the resulting segmented spots are classified into two classes: „true spots“ and „false positives“.

Netten et al. [Netten et al., 1996] used cell nucleus in slides of lymphocytes from a blood culture and introduced an automatic counting of spots. Their methods consists of three steps: 1) filtering to suppress noise and applying a global threshold, 2) detection of nuclei in using morphological filters, 3) segmentation of hybrization spots within the nucleus using a nonlinear filter and top-hat transform.
Their performance in spot detection is acceptable, but often this method yields segmented spot regions that contain mislabeled pixels near the borders of the spots.

Lerner et al. [Lerner B et al., 2007] proposed a FISH image classification system which is not limited to dot-like signals and provides a methodology for segmentation and classification of dot and non-dot signals. This method is based on the properties of in-focus and out-of-focus images captured at different focal planes. They were used initially for the classification a neural network (NN)-based algorithm (well described in [Hastie et al., 2009]) and later provided a Bayesian classifier instead of NN, to avoid dependency on a large number of parameters and the NN architecture settings.

7.3 Colocalization Measuring Methods

As it is well-known, one of the main tasks of modern biology is localizing subcellular objects within the cell in an accurate way. It is important to note that cellular functions depend on the interaction of subcellular structures, where the spatial proximity and correlation between the interacting structures are crucial [Helmuth et al., 2010]. One of the most commonly used methods for this purpose is the colocalization analysis.

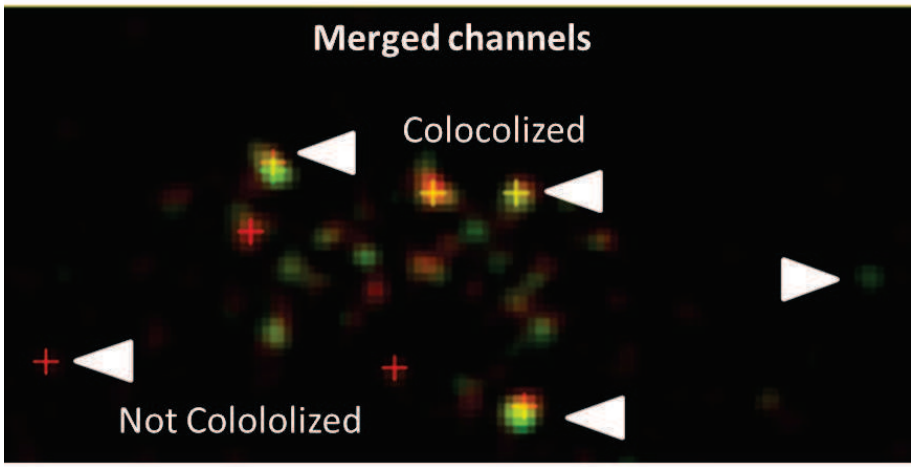


Figure 7.30: Colocalization of two fluorescent signals. Image data from green and red fluorescent proteins from the same view are merged. Colocalization overlay of panels showing areas with colocalization and no colocalization

As shown in Figure 7.30, the colocalization analysis determines whether subcellular structures are located at the same physical position where they can interact with each other or not. Thus the colocalization analysis of fluorescence is performed in paired or more images based on superimposition (merging) of images. In case of an image pair, typically a green channel is analyzed with a red channel to find out in which pixel they are colocalizing. In the context of digital imaging this means that the colors emitted by the fluorescent molecules occupy the same pixel in the image. First initial approaches were based on manual visual inspections which lead to inaccurate and error-prone results.

The quantitative colocalization measuring methods can be classified into:

1. **Intensity Correlation Coefficient-Based ICCB**:
   ([Maanders et al., 1993], [Costes et al., 2004], [Hansen, 1990], [Li, 2004])

2. **Object-based schemes**:
   ([Boutte, 2006]; [Lachmanovich et al., 2003]),

which are described in more details in next sections.

7.3.1 Intensity Correlation Coefficient-Based (ICCB)

The *intensity correlation coefficient-based (ICCB)* is a global statistic approach and relies on individual pixels assuming that each pixel is part of the image and not part of a unique object. They measure the colocalization degree by computing the correlation coefficient (e.g. *Pearson's correlation coefficient* [Soper et al., 1917] and *Manders' coefficient* [Maanders et al., 1993]) or an extension of them, the *Costes' approach* [Costes et al., 2004].

The ICCB tools investigate the correlation between two channels based on the relationship between fluorescence intensities of both channels. In other words, for each pixel coordinate the corresponding intensity in first channel is compared with the intensity in the second channel. Closer intensities lead to higher correlation value. The most used correlation parameters are based the Pearson's coefficient and the Manders' coefficient, which are described further below.

The relevant ICCB approaches are already brought together and available in the community. Bolte introduces *JACoP* [Fabrice P. Cordelieres, 2006] a plugin for ImageJ [Schneider et al., 2012] including most important ICCB tools which allows users to compare various segmentation methods based on ICCB.

The main disadvantage of the ICCB is that no spatial exploration of the colocalized channels is possible [Bolte and Cordelieres, 2006]. Furthermore this approach is affected by noise, since two very similar noisy regions in an image pair can lead to increase the correlation value.

Correlation Analysis based on Pearson's Coefficient

Pearson's correlation coefficient (PCC) is a well-defined and accepted measurement for describing the degree of overlapping (colocalization) between image pairs. It is a value between -1 and 1, where values close to zero indicate no colocalization and near 1 represent colocalization with -1 standing for negative correlation.

Assuming that two channels Red (R) and Green (G) of an image data are given. Pearson's correlation value is calculated according to the following formula:

$$R = \frac{\sum_i (R_i - \overline{R}) \cdot (G_i - \overline{G})}{\sqrt{\sum_i (R_i - \overline{R})^2 \cdot (G_i - \overline{G})^2}}$$

where R is signal intensity of pixels in the first channel, G is signal intensity of pixels in the second channel, further $\overline{R}$ and $\overline{G}$ are average intensity of first and second channels respectively.

The Pearson's Correlation between image pairs can be also visualized using a scatter plot. Figure 7.31 shows the scatter plot containing a set of points appearing as a cloud and a line in the middle. Each point represents for a certain pixel the intensities of both channels against each other.

The Pearson's coefficient has on the one hand the advantage that it is not dependent on a constant background and on image brightness, but on the other hand it is not easy to interpret and affected by addition of non-colocalizing signals.

Correlation analysis based on Manders' coefficient

Analogical to the Pearson's coefficient the Manders' overlap coefficient is based on intensity correlation coefficient. It excludes the average intensity values of the channels in the mathematical computation. The Manders' coefficient varies from 0 to 1 and reflects non-overlapping and 100% colocalization between images, respectively.

Two different values M_{red} and M_{green} are calculated as follows:

$$M_{red} = \frac{\sum_i R_{i,coloc}}{\sum_i R_i} \qquad M_{green} = \frac{\sum_i G_{i,coloc}}{\sum_i G_i}$$

where $R_{i,coloc} = R_i$ if $G_i \succ 0$; $G_{i,coloc} = G_i$ if $R_i \succ 0$.

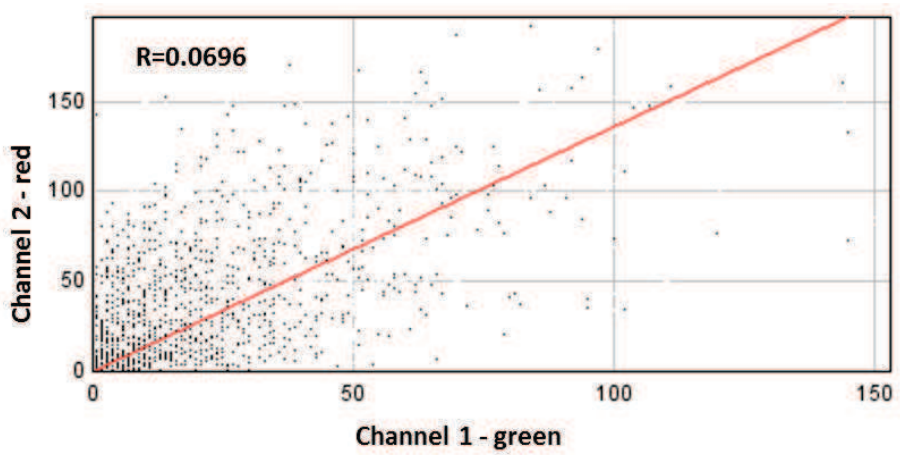


Figure 7.31: Scatter plot of Pearson's Correlation analysis. For each pixel the x-coordinates represents the intensity of the first channel (e.g. green channel) and y-coordinates (e.g. red channel) of the second channel. The red line represents the case of complete colocalization.

M_{red} is defined as the ratio of the summed intensities of pixels from the green image for the intensity in the red channel is above zero to the total intensity in the green channel and M_{green} is defined conversely for red, thus in general the proportion of overlap of each channel with the other. I.e. M_{red} is the sum of the intensities of red pixels that have a green component divided by the total sum of red intensities. Thus referring to fluorescence image, the coefficients M_{red} and M_{green} are proportional to the amount of fluorescence of colocalizing objects in each component of the image, relative to the total fluorescence in that component.

For example the interpretation for values $M_{red} = 1$ and $M_{green} = 0.65$ can be interpreted as follows: 100% of red pixel intensities colocolize with green channel, but only 65% of green pixel intensities colocolize with the red channel.

7.3.2 Object-based Approach

The introduced ICCB tools have the main disadvantage that the spatial exploration of the colocalized channels is not considered. The ICCB method, as described previously, considers that each pixel is part of the image and not part of a unique object.

The correlation analysis needs to take into account, both the size and the form of the colocolized object in a three-dimensional context, since as it is known, biological objects are three-dimensional structures varying in size, form and intensity distribution. The goal of the object-based colocalization analysis is to investigate the overlapping degree of individual structure of image pairs. Thus in contrast to the ICCB, the structures of images should be detected and defined beforehand.

In general two objects are typically considered as colocalized (Figure 7.32), if both objects overlap to a certain degree or if the distance between both centers of mass is below a certain threshold [Boutte, 2006]. An appropriate value for the distance threshold is the optical resolution. I.e. two objects colocolize if the distance between their centroids is lower than the optical resolution.

Prior to the colocalization analysis, the 3D image should be investigated to find three-dimensional connected components (mostly, commonly known as spot segmentation step) which is described in Section 8.1. The object-based methods can be performed among others either by *Overlap approach* or *Distance approach* (see Figure 7.32) which are described in next section.

Distance approach

The *distance approach* is based on the distance between two reference points of objects. First all objects should be segmented, for each of them the centroid with their x-,y- and z-coordinates is determined. The coordinates of the center points are decisive for the colocalization measurements. As depicted in Figure 7.32 (left), if the distance between the centers of two different structures are below optical resolution, this means that these structures colocalize [Boutte, 2006].

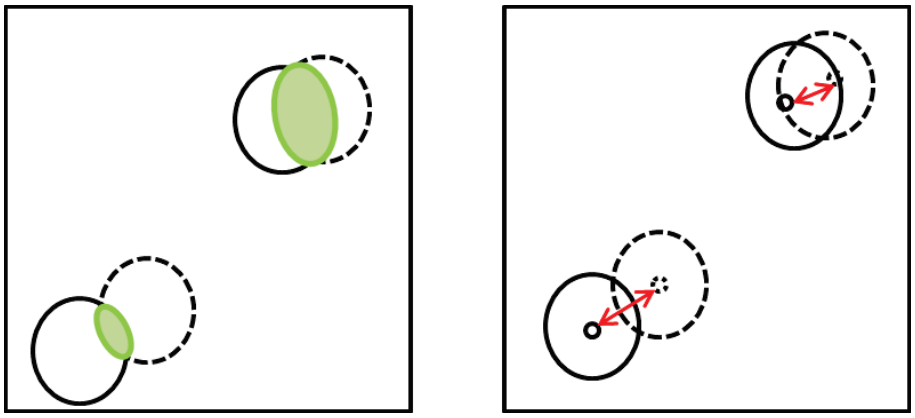


Figure 7.32: Sample illustration of object-based colocalization approach.
The overlap approach (left) and the distance approach (right) are sketched. The objects on the top right are samples for colocolizing because either the overlapping degree is enough (left) or the distance between their centers are less than a specific threshold (right). Two samples below on the left are samples of non-colocalization.

Based on the distance measure, Lachmanovich et al. introduced an approach for investigating the colocalization of two objects [Lachmanovich et al., 2003]. According to their proposal, two objects colocalize if their distance is lower than a specified distance t. More formally:

$$\delta^t = \frac{1}{N} \sum_{i=1}^{N} I(d_i < t)$$

where $I(\ \cdot\ )$ is the indicator function and t is the specified distance and defined as:

$$I(\ \cdot\ ) = \begin{cases} 1, & \text{for } \ d_i < t \\ 0, & \text{otherwise.} \end{cases}$$

Furthermore, [Lachmanovich et al., 2003] defines the degree of colocalization based on the percentage of objects in the first channel colocalizing with objects from the second channel, divided by the total number of all objects from the first channel. Since the number of objects can differ between two channels, the measurement should be set to count the objects from the channel with fewer objects and search for objects from the channel with more objects [Lachmanovich et al., 2003].

Overlap approach

Presenting objects by their centroids without considering other object properties (e.g. size or form) may lead to under-estimation of colocalization due to differences of intensity distribution and in size of the objects. For this reason Lachmanovich et al. [Lachmanovich et al., 2003] introduced an overlap approach where the area of objects are considered. As depicted in Figure 7.33 two objects of two channels colocalize if the centroid of an object from the first channel (e.g. green) falls into the area covered by an object of the second channel. The colocalization measure or overlapping degree is calculated by counting the number of objects in the first channel matching second channel object areas and reverse. Additionally to evaluate the measured colocalozation degree, it will be compared with random distribution image pairs.

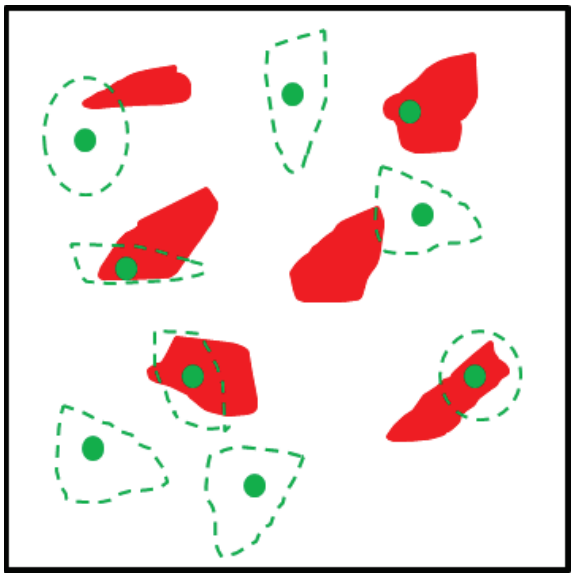


Figure 7.33: Illustration of overlap object-based approach. Merged view of centroids of the green image and object areas of red image. If the green centroids overlap the object area of the red channel, then these two objects colocolize with each other.

The extension of this approach as depicted in Figure 7.32 (left) which considers the area of both objects to determine how large the overlapped area between them is. In this case an appropriate threshold should be defined to have a decision rule.

7.4 Colocalization Analysis based on Spatial Point Processes

As mentioned in Section 7.2.2, several algorithms have been proposed for quantification of colocalizations in multichannel 3D Fluorescent images. The main drawback of the most proposed methods is that essential spatial characteristics of the colocalized objects such as cellular context, their surrounding distribution and the expected occurrence probability are not considered. Moreover the interpretation is often imprecise due to the lack of information whether the observed colocalization result is statistically significant and if the observed values occurred by chance or not [Fletcher et al., 2010]. Analyzing the colocalization without considering the spatial interaction of structures under-exploits the information [Helmuth et al., 2010]. Consequently, for an improved and more extensive analysis, spatial point process methods are required. This kind of analysis provides the comparison between the spatial organization and distribution between an arbitrary subcellular object with a reference object [Andrey et al., 2010].

Due to the drawbacks of colocalization analysis irrespectively of their spatial characteristics, some methods are developed to investigate the spatial analysis. All spatial organization studies require a preliminary segmentation of objects in the study area, which after that the objects are represented by a set of identifiable points (in our case three-dimensional). For an overview of developed detection and registration methods see [Bolte and Cordelieres, 2006].

There are already some three dimensional spatial point process approaches introduced in nuclear biology like [Gué et al., 2006],[Mahy, 2002],[Ronneberger et al., 2008],[Shiels et al., 2007]. Additionally for characterizing transcription factors, there are rarely spatial statistics based on the distance function studies and quantitative measures ([McManus et al., 2006],[Young, 2004], [Noordmans et al., 1998]). Recently a spatial analysis has been applied for investigating five very diverse populations of interphase animal and plant nucleus based on different distances measurements [Andrey et al., 2010].

In order to study nuclear organization, ([Helmuth et al., 2010]) proposed to measure the colocalization in the context of spatial point pattern analysis. The spatial analysis is used to describe the underlying structure and to measure the interactions between subcellular structures. Before dealing with spatial analysis,

first we introduce the theory of spatial point process in the following sections. In particular, the R package spatstat [Baddeley and Turner R., 2005] for the analysis of spatial point patterns was employed.

7.4.1 Introduction to Spatial Point Processes

In this section, some basic notation as well as the most important concepts regarding spatial point processes are introduced. These aspects are discussed extensively in a variety of books, such as [Illian et al., 2008],[Diggle, 2003],[Møller and Waagepetersen, 2004] and[Commenges, 2011]. The theory and introduction of spatial point processes in one-dimensional space was published by [Daley and Vere-Jones, 1988].

A spatial point process consists of a finite or infinite random set of points, whereas each point is a realisation of a real object such nests, trees, galaxies, earthquake epicenters and in our case cell nucleus). In principle the points can be situated in quite general spaces such $\mathbb{R}$, $\mathbb{R}^2$ or $\mathbb{R}^3$. The definition and introduction are based on $\mathbb{R}^2$ space for a better visualisation and comprehension. Further, the analysis in next chapter is based on a $\mathbb{R}^3$ space. In general, the spatial point process deals with analyzing of the geometrical structure of point patterns formed by objects which are derived from various applications [Illian et al., 2008].

In this chapter, after introducing the spatial point process, the existing approaches are used in the context of nucleus biology research with the aim to develop methology that is suitable for this purpose. Following definitions from spatial point process (all quoted from [Illian et al., 2008]) are essential for dealing and understanding the colocalization analysis based on spatial statistics:

Points and marks: In spatial point processes, *points* and *marks* are used in order to represent objects. The location of the objects are characterized by points and for additional information about each point (e.g. size, type or form) marks are used. The mark represents an attribute of the point, which can be distinguished between categorial marks (e.g. cell type) and contnious (e.g. cell size or diameter).

The most efficient and central part of spatial statistics is the spatial point process, which is described as follows [Illian et al., 2008]:

Point pattern X: Spatial point processes are „stochastic models of irregular point patterns“. A set of points conforms the definition of a random point process, where a point process X_i is defined as a set of 3D points x_i with a random number and location of the points:

$$X = \{x_1, x_2, \ldots, x_n\}, x_i \in \mathbb{R}^d, n \geq 0, d = 3$$

In other words a specific observed point x_i can be considered as a realisation of a random point pattern X. The goal is to describe the distribution and further properties parametrically and estimate the parameters based on an observed point pattern X.

Observation window W: As depicted in Figure 7.34, we assume that the point process X extends throughout a 2D or a 3D space, but is observed only inside a region of interest which is defined as an observation window W. It is important to know the observation region W, since we need to know where points are not observed. In other words, the observation window is the area for which data are available.

An essential question in this context is how to deal with critical points, which are completely or partial located at the edge of the defined window W? This question is dealt by *edge-correction method* (described in [Illian et al., 2008]).

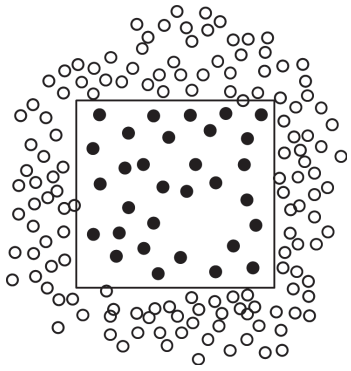


Figure 7.34: An illustration of the study region (window). The standard model assumes that the point process extends throughout a 2D or 3D space, but is observed only inside a region W

The intensity λ: The intensity or point density λ is the first essential value in a point process and defined as the average density of points in a specific area. In other words, it is the expected number of points per unit area or volume. The intensity may be constant over the whole observation area (*uniform* or *homogenous*) or may vary from location to location (*non-uniform* or *inhomogeneous*). These characteristics are described in next sections.

Spatial stationary: Consider a point process $X = \{x_1, x_2, ..., x_n\}$ and the translated point process

$$X_y = \{x_1 + y, x_2 + y, ..., x_n + y\}$$

which obtained by shifting all points of the process X by the same vector y. A point process is stationary if for any subregion of the observation window, the statistical properties (e.g. the distribution) are not changed and are stable against rotation and shifting. Hence, in this case the points process X is statistically invariant under translations [Illian et al., 2008].

Distance measure: For measuring the distance between two three-dimensional points x_i and x_j, we use the 3D Euclidean distance defined as:

$$d(x_i, y_i) = \|x_i, y_i\| = \sqrt{(x_1 - y_1)^2 + (x_2 - y_2)^2 + (x_3 - y_3)^2}$$

Colocalization measure are typically based on two sets of objects (e.g. green vs. red channel), which are posed in our case three-dimensional location vectors of points:

$$X = \{x_i\}_{i=1}^{N} \text{ and } Y = \{y_i\}_{j=1}^{M}$$

Counting measure: Assuming $W \subseteq \mathbb{R}^d$ is the observation area of the spatial point process X, for each subregion of W the so-called *Borel set* $B \subseteq \mathbb{R}^d$, its area is denoted by $\nu(B)$. The counting measure $N(B)$ denotes the random number of points in a Borel set. Additionally, for disjoint sets $B1$ and $B2$, the sum of the point counts is defined as:

$$N(B_1 \cup B_2) = N(B_1) + N(B_2)$$

After introducing the basics of spatial point process, the next section describes most important spatial distributions.

7.4.2 Point Process Distributions

Spatial point processes are used in order to describe and model spatial structures, which are formed by the locations of individuals in space. The spatial point processes can be primarily distinguished between following characteristics:

- Complete spatial randomness (homogenous Poisson point process or uniform)
- Regular process
- Clustered process
- Inhomogeneous Poisson process (non-uniform)

Homogenous Poisson point process

The homogeneous Poisson process is the most important point process model and represents the case of complete spatial randomness. It can be used as a *null model* in order to check if an arbitrary real point distribution has any systematic structure or is distributed randomly. The goal is to find out if the point pattern is more likely to be random and to asses this formally with statistical methods.
If the point process is known to be homogeneous, then the intensity λ is constant over the whole space. It describes the mean number of points in a unique square, which is defined as follows:

$$\lambda = \frac{N(W)}{\nu(W)}$$

where $N(W)$ denotes the number of points in study area W and $\nu(W)$ is the size of the study area.

The homogeneous has following essential characteristics:

- The number of points in any subregion B of any size is independent for arbitrary k and follows a Poisson distribution: $N(B) \approx Po(\lambda\nu(N))$
- The counts $N(B_1)$ and $N(B_2)$ are independent for disjoint sets B_1 and B_2.
- The homogeneous Poisson process is stationary.

Figure 7.35(b) depicts a simulated example of a Poisson point process on a unit square. As it is shown the points are randomly distributed over the region, but the expected number of points over the equal subregions is the same. The homogeneous Poisson process is also referred to as *Complete spatial randomness (CSR)* which has a central role in point process statistics. CSR is described as follows: Any point is equally likely to occur at any location and the position of any point is not affected by the position of any other. CSR which is a homogeneous Poisson distributed can be used as a null, benchmark or reference model to distinguish between point patterns exhibiting aggregation and repulsion (described in next sections).

In order to determine whether a colocalization value was significant and did not occur by chance, it should be compared in terms of point distribution to the CSR[Fletcher et al., 2010]. CSR is a standard model where the points are randomly distributed and followed approximately a homogeneous Poisson process over the study area ([Gelfand et al., 2010]). For known intensity λ, all types of distributions of the process can be determined as follows:

$$P(k \mid \lambda) = \frac{\lambda^k}{k!} \cdot exp(-\lambda)$$

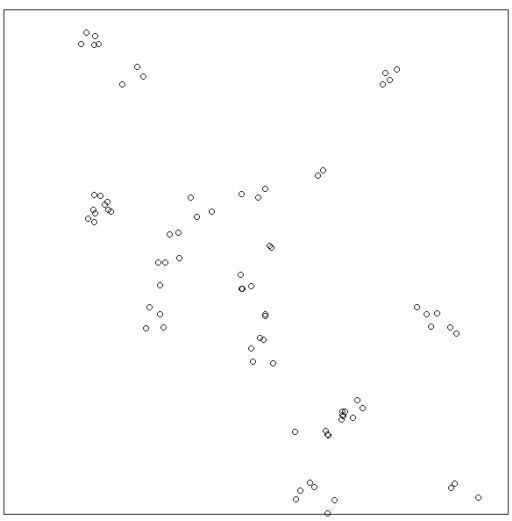
Assuming there are k randomly located points in a study region, the assumption of spatial randomness means that any point has an equal probability to occur at any position and their position does not affect one another. In statistical terms, the set of all points are assumed to be statistically independent [Illian et al., 2008].

Clustering and regularity

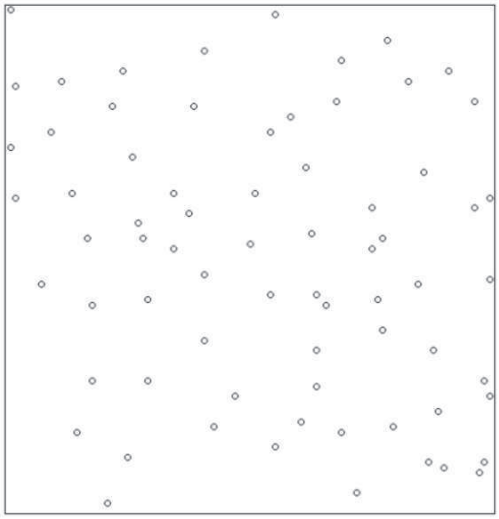
Two further variation of points processes occur, when points repulse or attract each other. Repulsion leads to *regular* patterns and the interaction yields *clustered* patterns. In the regular process, the distance from an arbitrary point to its nearest neighbour is typically large and further from each arbitrary location in the observation window to the next nearest point of the process is roughly the same. In other words, every point is as far from all of its neighbors as possible. Figure 7.35(c) shows a simulated example of such a regular pattern, next to a simulated example of a Poisson process.

The opposite type of interaction is repulsion, which yiels clustered or aggregated patterns. Here, many points are concentrated close together, large areas that contain very few or any points and the distance from an arbitrary point to its nearest neighbour is typically small. Further, the distance from each arbitrary location in

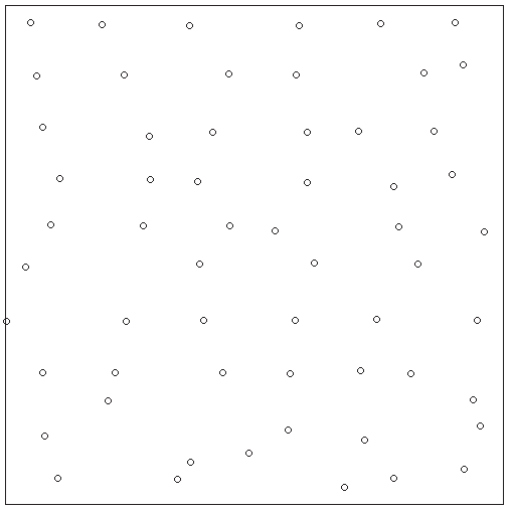
the observation window to the nearest point of the porcess is typically large. A simulated example of such a clustered pattern is depicted in Figure 7.35(a). As one can imagine, regular patterns are not relevant for the colocalization analysis application and will threfore play a minor role in the following parts.



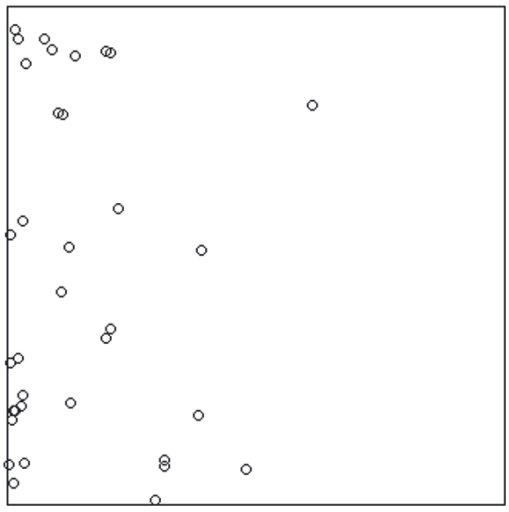
(a) clustered process



(b) Poisson process (homogeneous)



(c) regular process



(d) inhomogeneous Poisson process

Figure 7.35: Simulated examples of a clustered process, a Poisson process, regular and inhomogeneous Poisson process.

Inhomogenous Poisson point process

Another deviation from complete spatial randomness is the inhomogeneous pattern, where its intensity is varying from region to region and is depend on the location s. Therefore, compared to the homogeneous Poisson point process, the formerly constant intensity λ is replaced by an intensity function $\lambda(s)$ whose values depend on the location $s \in W$.

Thus the expected intensity measure for all regions B which is the number of points from X falling in region B can be written as:

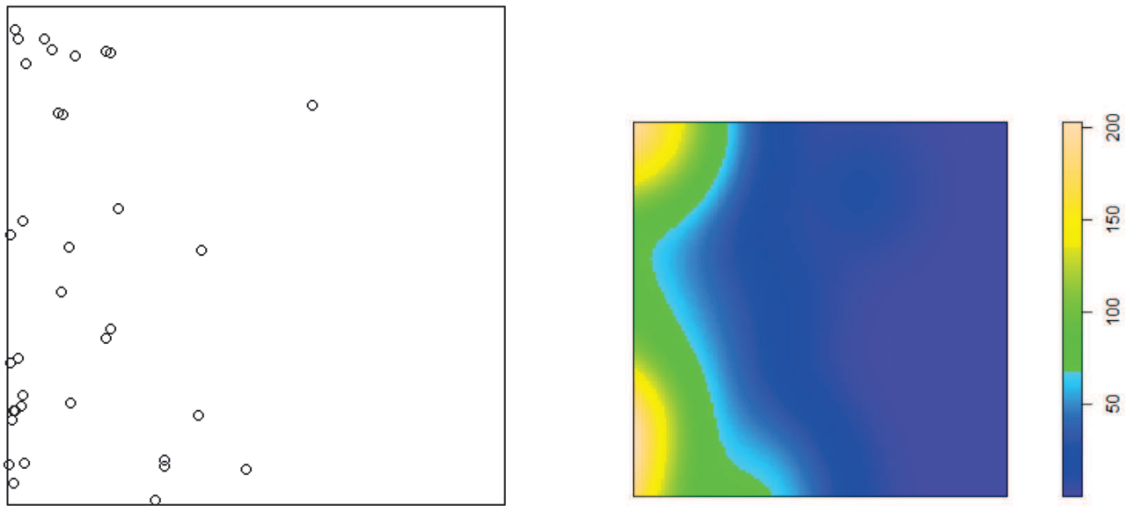
$$\Lambda(B) = E[N(X \cap B)] = \int_B \lambda(s)\ du$$

where $\lambda(\ \cdot\ )$ is called the *„intensity function“*.

Inhomogeneous point patterns with an intensity depending on the location are observed quite frequently in nature and technology. For example, the number of trees per unit area in a forest is not constant, but rather it depends on environmental conditions. The assumption of an inhomogeneous Poisson point process implies that beyond spatial variation in the intensity function, there is no stochastic dependence between observations. The expected number for each Borel set B is defined as:

$$P(N(B)) = \frac{\Lambda(B)^k}{k!} \cdot exp(-\Lambda(B))$$

where $\Lambda(B) = \int_B \lambda(s)\ du$.



(a) inhomogeneous Poisson process

(b) intensity function

Figure 7.36: Simulated examples of inhomogeneous Poisson processes and their intensity functions.

7.4.3 Spatial Statistic Approaches

Spatial analysis consists of a set of mathematical and statistical operations and deals with the finding of patterns in spatial datasets. Spatial patterns are used in order to understand the underlying structure.

The following three goals are pursued by using spatial analysis:

- Characterization of the point pattern properties formally with statistical methods
- Determining if there is a tendency of points to exhibit a systematic pattern over an area as opposed to being randomly distributed
- Confirmation whether a spatial pattern found in visual analysis is statistically significant.

Point patterns analysis can be performed in two ways:[Bailey and Gatrell, 1995]

1. Exploratory spatial analysis: Describing the intensity and inter-point distance of the points.
2. Confirmatory spatial analysis: Hypothesis test methods.

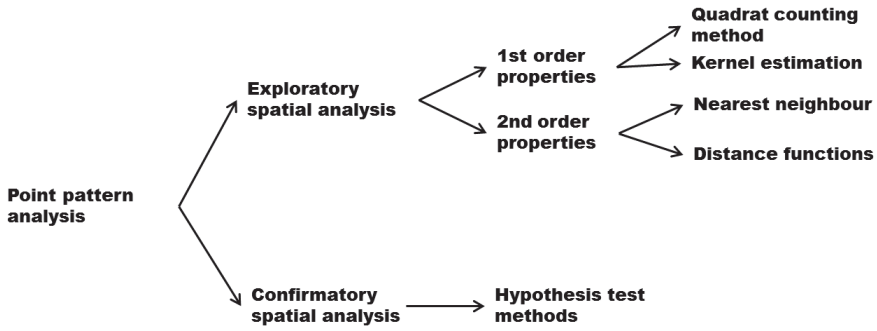


Figure 7.37: An overview of point pattern analysis methods. [Bailey and Gatrell, 1995] classify the methods first into exploratory and confirmatory spatial analysis methods.

1. Exploratory approaches

The exploratory approach analyses either the intensity of point pattern varies over an area (1st order) or estimates the presence of spatial dependence among points (2st order). As shown in Figure 7.37 the exploratory pattern can be subdivided into *1st-order* properties (e.g. studying the point intensity λ) and *2nd-order* properties (e.g. focusing on the inter-point distance in a study region).

Typically approaches for measuring of 1st-order properties are the *Quadrat counting method* (in homogeneous case) and *kernel estimation* (in non-homogeneous case). Further for 2nd-order properties the *Nearest neighbor distance* measure and some distance functions such *Ripley's K-function and L-function* and *pair correlation G-function* are commonly used which are described below.

First-order characteristics

The first order characteristics describes the density of points through the observation space, which can be either constant over the area or non-constant. The intensity λ of a point pattern which belongs to the first-order characteristic can be estimated either parametrically or non-parametrically. In this section three commonly used approaches based on the first-order properties are introduced. All of them pursue the same goal to find out if the underlying distribution is completely randomly distributed (CSR) or not. For this investigation it is important to note that under the CSR assumption, the points are homogeneous Poisson distributed [Illian et al., 2008].

I. Quadrat Counting Method: Quadrat counting another very common approach to estimate the intensity function in a non-parametrically way.
Under this approach, first the sampling window W is divided into m rectangular subregions $B_1, \ldots, B_m$ in form of quadrates or squares of same size. For each subregion the counts N_i indicate the number of points falling into subregion B_i.

Based on the known formula for estimating the point density in a sampling window W, where $\bar{\lambda} = \frac{N(W)}{\nu(W),}$, the Poisson cell-count distribution can be estimated by:

$$Po[N_i = k \mid \bar{\lambda}] = \frac{(\bar{\lambda}a)^k}{k!} \cdot exp(-\bar{\lambda}a)$$

where the number of the points in each subregion is independent and they are samples from this Poisson distribution with the parameters k and λ.

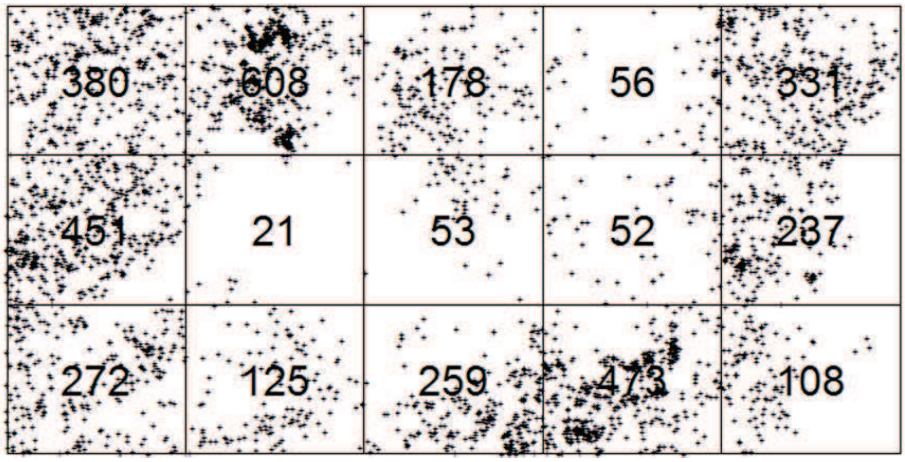


Figure 7.38: A sample of quadrat counting method. The study region W is divided into $m = 15$ subregions and each region the number of points are counted.

In case of CSR (Complete Spatial Randomness) we assume that the frequently number of points in each region will follow a Poisson distribution. To describe the distribution type and to test the CSR, the very simple *index-of-dispersion test* should be calculated and compared with a Poisson distribution in case of CSR.

The index of dispersion I is defined by [Illian et al., 2008]:

$$I = \frac{(m-1) \cdot \sigma^2}{\bar{N}},$$

where m is the number of quadrats, $\bar{N}$ is the mean number of points per quadrat, and σ^2 is the sample variance of the number of points per quadrat. The index of dispersion is used as a rough measure of dispersion versus clustering.

Assuming that after subdividing the study region in m subregions and counting the number of points in each subregion, the mean and variance of these numbers are $\overline{n}$ and σ^2 respectively. In case of CSR (Poisson distribution) the variance and the mean are the same, we expect a index-of-dispersion around 1. For clustered distribution, the variance is relatively large and we expect $I > 1$.

Following cases might occur:

- $I < 1$**:** There is too little variation among quadrat counts, suggesting possible „dispersion“.
- $I > 1$**:** There is too much variation among quadrat counts, suggesting possible „clustering“ rather than randomness.
- $I \approx 1$**:** Ratio values near to 1 indicate a „randomness“, thus it follows a CSR.

The choice of quadrat size is critical. If the quadrat size is too small, they may contain only a couple of points and many quadrates have zero points. If the quadrat size is too large they may contain too many points and many quadrates would have a similar number of points [Illian et al., 2008].

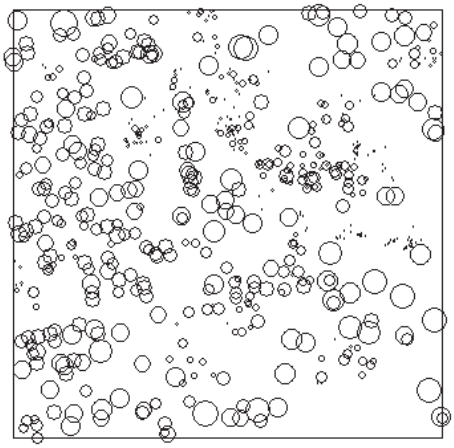
II. Kernel Estimation: This approach also belongs to the exploratory analysis and measures the 1st-order property in a non-parametrical way. The idea is to calculate the density of events within a specific search radius around each event. This approach is applied in inhomogeneous cases, where the density varies from location to location. A moving three-dimensional function which is called the *kernel* of a given radius meets each point of the study area. The kernel is used to weight the area surrounding the point proportionally to its distance to the reference point. After summation of these individual kernels for each study region a smoothed surface is produced.

Assuming n is the number of points, τ is the specific search radius (bandwidth), the parameters s_i with $i = 1, ..., n$ are individual locations of n points in study region and s is one location in study area. Then the kernel estimation which can be thought as a linear filter is a function of convolution and defined as:

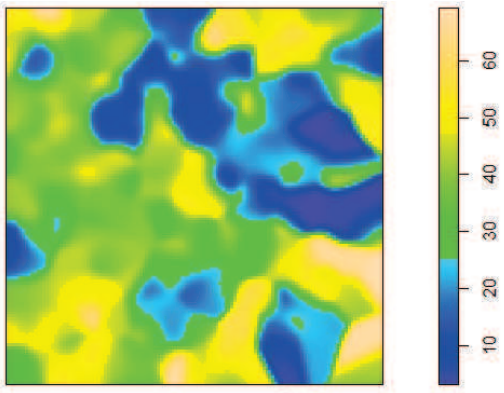
$$\lambda_\tau(s) = \sum_{i=1}^{n} \cdot \frac{1}{\tau^2}(\frac{s - s_i}{\tau})$$

with a specific kernel κ [Illian et al., 2008] and [Baddeley, 2006].

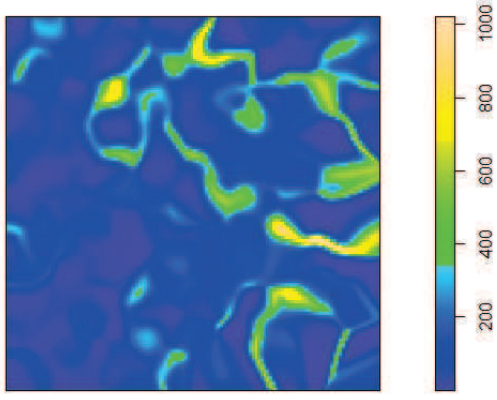
For visualization and better comprehension the „Longleaf Pines point pattern“ introduced by [Platt et al., 1988] is used. This data set describes the locations and sizes of Longleaf pine trees. Since the size is considered, it is a marked point pattern. Further, the data record the locations and diameters of 584 Longleaf pine (Pinus palustris) trees in a 200 x 200 metre region in southern Georgia. Figure 7.39 shows first the marked locations of the 584 trees with the appropriate intensity function. Figure 7.39 (c) illustrated the $\lambda_\tau(s)$ is determined as a result of smoothing based on kernel estimation.



(a) Point locations



(b) Intensity function



(c) Smoothed intensity function

Figure 7.39: Simulated examples of kernel estimation based on [Platt et al., 1988]

Second-order characteristics

The second-order characteristic describes the correlation in form of distances between the points and considers the expected number of points at specific distances from each point. Further apart from the nearest neighbor approach, where a constant threshold is used, the indices are based on functions. I.e., exploring the spatial distribution depending on different input values such as distance. In the next section the three most commonly used measures, the *Ripley's K-function*, *Besag's L-function* and the *pair correlation function G* are described. These approaches prove the CSR characteristic like first-order approaches, but based on the distance between the points and not their distribution.

I. Nearest Neighbour Analysis: This approach belongs to the category of exploratory approaches based on 2nd-order properties. It looks at distances between points. Assuming in a given point pattern $X = \{x_1, \ldots, x_n\}$ the *nearest neighbor distance (nn-distance)* for any point x_i to all other points in X is given by:

$$d_i = d_i(X) = \min_j \{d(x_i, y_j) : x_j \in X, i \neq j\}.$$

To describe the *nearest-neighbor index* $\bar{d}$, the average of these nn-distances provides a direct measure in point pattern:

$$\bar{d} = \frac{1}{N} \sum_{i=1}^{N} d_i,$$

where d_i is the distance to the nearest neighbor from point i and N is number of points.

Assuming $b(x, r)$ standing for disc of radius r centered at x, the nearest neighbour distance can be defined as a function of the distance r, assuming a Poisson point process is given, as follows:

$$D(r) = P_o(N(b(o, r) \setminus \{o\}) > 0)$$

for $r \geq 0$ describes the random distance from a initial point o to its nearest neighbour.

Further, for a homogeneous Poisson point process, the nearest neighbour dustance distribution function is [Illian et al., 2008]:

$$D(r) = 1 - exp(-\lambda \pi r^2).$$

Large values indicates clustering, whereas smaller values suggest regularity. Hence, after standarization (dividing by the Poisson process value) following cases may arise:

- $D(r) < 1$**:** Small value for w indicates „clustered“ points.
- $D(r) \approx 1$**:** Values near to 1 means that the points are randomly distributed
- $D(r) > 1$**:** Values greater that 1 indicate that points are dispersed.

One of the main weakness of nearest neighbor approach is that the distance measurement is limited to the next nearest neighbor without considering how other points are located and dispersed. Therefore, we have to consider other spatial properties of the point process and cannot describes the behavior of the process at large distances .

II. Pair correlation function G(r) (distance function): One of the simplest measures of the dispersion is the pair correlation function $G(r)$ based on nearest neighbors, which investigates if the next neighbor is within a specific distance r.

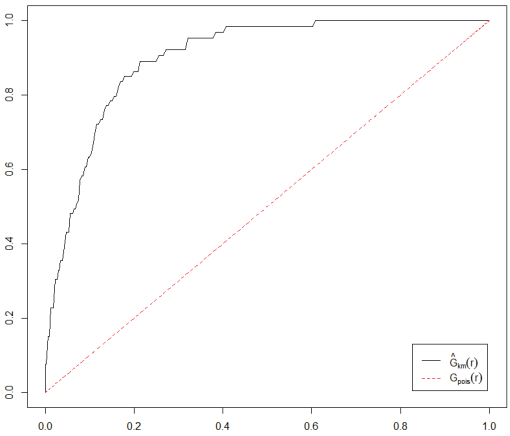
For various distance values $r \in [0;\ r_{max}]$ the corresponding values $G(r)$ are calculated as the number of neighbors with a distance smaller than r divided by the total number of points in the study area:

$$G(r) = \frac{\sum_{i=1}^{m} I_r(d_i)}{N(W)}$$

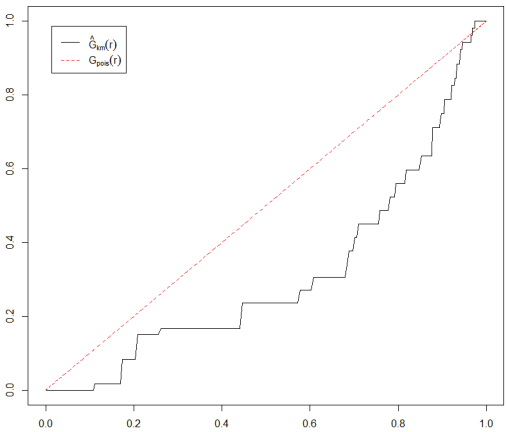
where $I(\ \cdot\ )$ is the indicator function defined by:

$$I_r(\ \cdot\ ) = \begin{cases} 1 & \text{, for } \ \mathrm{d}_i < r \\ 0 & \text{, otherwise.} \end{cases}$$

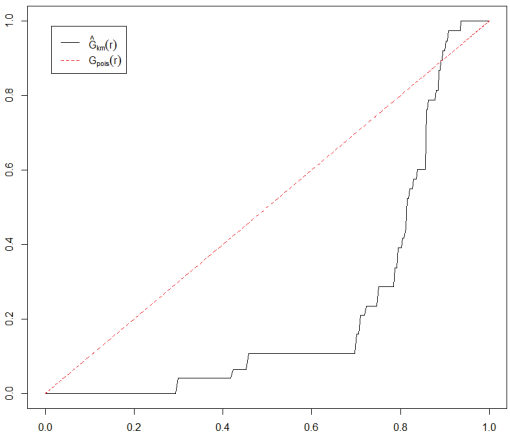
It returns 1 if the minimum distance of two points is smaller than r otherwise 0. The shape of G-function tells us the way the events are spaced in a point pattern. In case of clustered pattern, the G-function increases rapidly at short distance, and in evenness pattern G increases slowly up to a distance where most points are spaced, then increases rapidly. Furthermore the statistical significance of assuming of CSR or departure from CSR can be evaluated using the *Monte-Carlo approach* introduced in 7.4.4



(a) clustered process



(b) Poisson process (homogeneous)



(c) regular process

Figure 7.40: Nearest-neighbour distribution function: The solid lines represent the estimated nearest neighbour distribution functions for the observed patterns, the dashed lines correspond to the theoretical nearest-neighbour distribution functions of a homogeneous Poisson process based on the introduced distribution types in Figure 7.35

III. Ripley's K-function K(r): The G-function is limited only by the nearest neighbor distance. The key idea of Ripley's K-function is to investigate various surrounding regions with different distances r (circle radius) from each point as shown in Figure 7.41. The K-function indicates the average number of other points found within a certain distance r from each point, therefore it is a function of the distance h [Ripley, 1976]. The K-function estimates the spatial dependence over a wider range of scales. The aim is like previous approaches to find out, if the distribution of points is regular, random or clustered.

The K-function describes the degree of spatial clustering at the scale represented by the distance parameter r. First a circle of radius r around each point x_i is constructed. Then for each point, the number of other points that fall inside this circle is counted. After that the sum of these values corresponds to $K(r)$ is added.

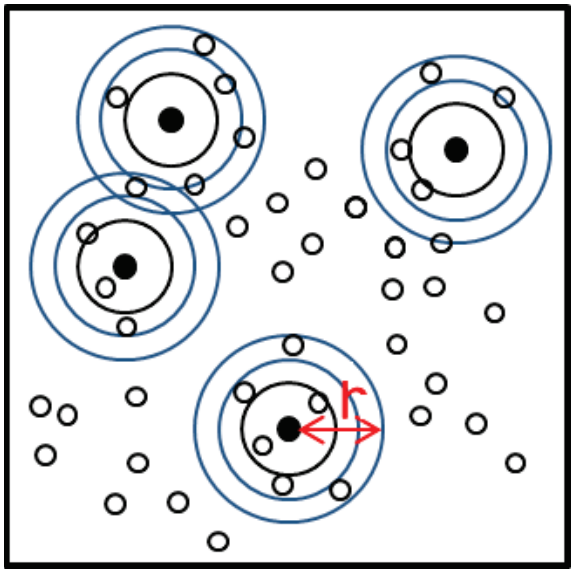


Figure 7.41: Calculation of K-function based on different distances. As depicted the K-function counts the number of points located in a circular region of radius r from every point in study region.

Note that the absolute number of points is directly dependent on the point density λ. It will of course change with different values of λ. Hence we should divide

it by λ to eliminate this obvious effect:

$$K(r) = \frac{\sum_i \sum_{j \neq i} \sigma_{ij}(r)}{n\lambda},$$

where r is the distance parameter and the *indicator* function σ_{ij} is defined as:

$$\sigma_{ij} = \begin{cases} 1 & \text{, if } |x_i - x_j| \leq r \\ 0 & \text{, otherwise.} \end{cases}$$

In order to evaluate the degree of spatial clustering of points, the distribution under CSR is considered. The expectation of K-function of points under CSR (homogeneous Poisson distribution) is:

$$E(K(r)) = \pi r^2$$

Comparing calculated $K(r)$ with its expectation enables to classify point distributions into one of three categories:

- $K(r) > \pi r^2$: Points are clustered.
- $K(r) \approx \pi r^2$: Points are randomly distributed.
- $K(r) < \pi r^2$: Points are dispersed.

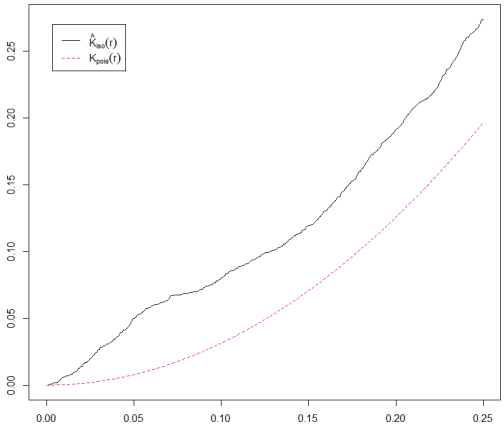
Further discussion to developed Ripley's approach lead to a suggestion by Julian Besag named „a slight modification to K-plots“ by showing a plot of $\sqrt{\frac{K(r)}{\pi}}$ againgst r [Besag, 1977]. This resulted in so-called Besag's ***L-function*** defined as:

$$L(h) = \sqrt{\frac{K(r)}{\pi}} \text{ for } r \geq 0.$$

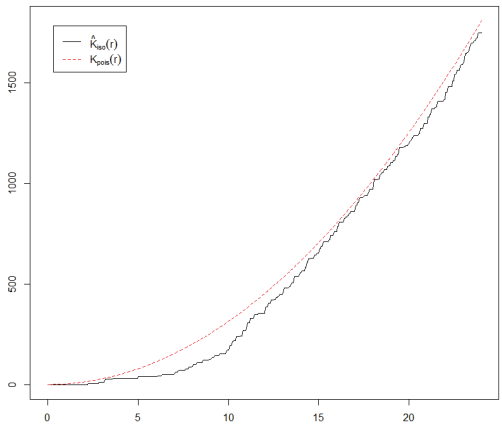
where this function can be compared with zero. Thus following three cases can occur:

- $L(r) > 0$**:** Points are clustered
- $L(r) \approx 0$**:** Points are randomly distributed
- $L(r) < 0$**:** Points are dispersed

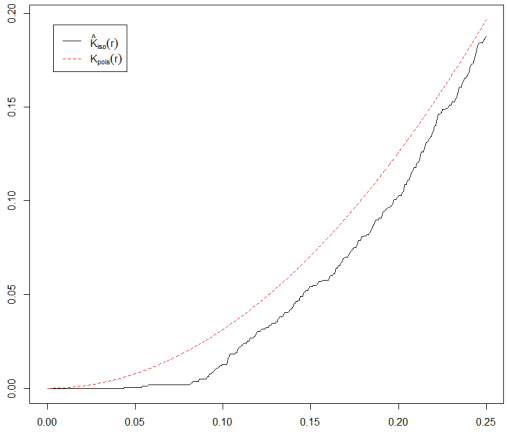
The L-function has the advantage, that it has an easier interpretation and visualization as the function is proportional to r. Figure 7.43 shows simulated examples of L-function based the different distribution types (introduced in Figure 7.35)



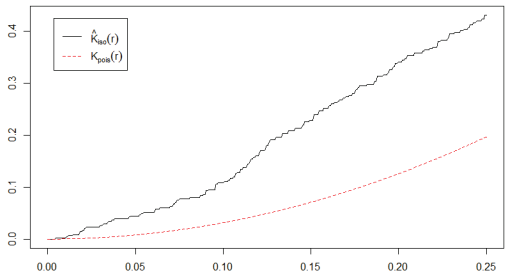
(a) clustered process



(b) Poisson process (homogeneous)

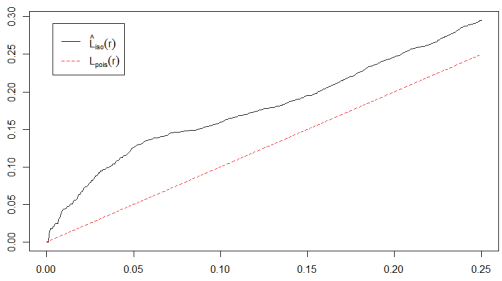


(c) regular process

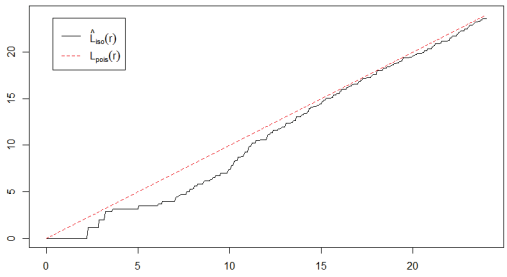


(d) inhomogeneous Poisson process

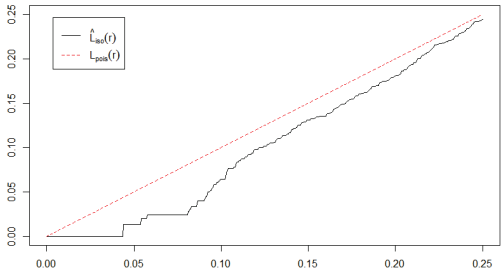
Figure 7.42: Ripley‘s K-function: The solid lines represent the estimated K-functions for the simulated patterns, the dashed lines correspond to the theoretical K-functions of a homogeneous Poisson process.



(a) clustered process



(b) Poisson process (homogeneous)



(c) regular process

Figure 7.43: Besag‘s L-function: The solid lines represent the estimated L-functions for the simulated patterns, the dashed lines correspond to the theoretical L-functions of a homogeneous Poisson process.

2. Confirmatory Spatial Analysis

The first introduced spatial point analysis type (exploratory approaches) can be extended by testing various statistical hypotheses about the point processes which are dealt in field of confirmatory analysis in order to make a decision with a statement about the reliability and significance of the point process. In other words, confirmatory approaches consists of statistical tests for the significance of spatial patterns in data, compared with the same characteristics of complete spatial randomness (CSR). It expands the exploratory approaches (descriptive measures) by indicating the reliability of the estimated value. This confidence interval reflects the significance of the calculated value. We want to found out if the observed spatial point pattern is significantly different from a complete spatial randomness.

In general, for examining an arbitrary point process for randomness (CSR), the hypotheses H_0 (Null hypotheses) and H_1(Alternative hypotheses) can be formulated as:

- H_0**:** Points are randomly distributed, following a homogeneous Poisson distribution.
- H_1**:** Points are spatially clustered or dispersed.

The general idea of CSR testing is as follows:
For a given data, a summary characteristics is estimated and compared with the relevant theoretical summary characteristic of a Poisson process. If there is a large difference between both characteristics, the Poisson *null-hypothesis* is rejected otherwise the hypothesis is accepted.

This topic will not be discussed in further detail here, since we don't use confirmatory methods for the colocalization analysis. Details can be found in the textbook [Illian et al., 2008]. Among all introduced confirmatory spatial analysis, the *Monte Carlo test* approach has been associated with the most suitable in our filed, wich is describes in the following section [Diggle, 2003].

7.4.4 Monte Carlo Test using Envelopes

A particularity of spatial point patterns is that simulation is an important part of the analysis as many important characteristics cannot be determined explicitly, at least not for more complex models [Diggle, 2003].

The main idea of the commonly used approach of *Monte Carlo*, which is used in various fields of science, is to solve a problem by generating suitable random numbers or simulations instead of analytical solving. It can be used for problems which are too complicated to be solved analytically [Metropolis and Ulam, 1949].

In cases where an arbitrary point pattern has to be investigated against CSR, a simulation of many randomly generated CSR curves (theoretical curves) is generated and the empirical curve of the point pattern is compared with this set of simulations.

The essential idea here is to simulate N independent simulations of the CSR inside the study region W. Compute from these simulated functions the lower and upper curves (called *envelopes*) of these simulated curves,

$$L(r) = min_j \hat{f}^{(j)}(h) \quad \text{and} \quad U(r) = max_j \hat{f}^{(j)}(h).$$

and compare the empirical distribution $\hat{S}(h)$ with the range of estimates $\hat{S}_i(h), i = 1, ..., N$ that lie within the envelopes $L(r)$ and $U(r)$. In other words we can find out the probability that $\hat{f}(h)$ lies outside the envelope [L(h),U(h)].

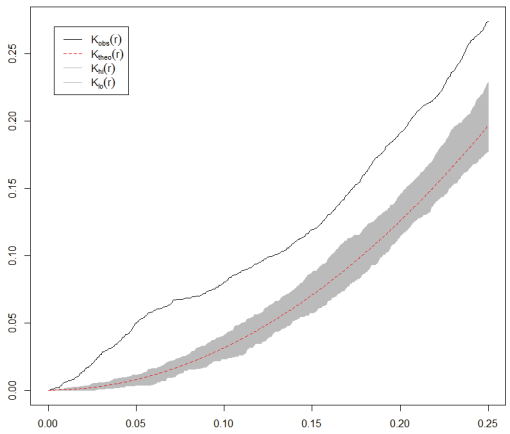
The statistical significance of any departures from CSR can be evaluated using simulated confidence interval, i.e. we simulate many (e.g. 1000) spatial point processes and estimate the function for each of these. Sort all the simulations and pull out the $5th$ and $95th$ $\hat{f}_i(h)$ values. Plot these as the 5% and 95% confidence intervals.

If the inequality

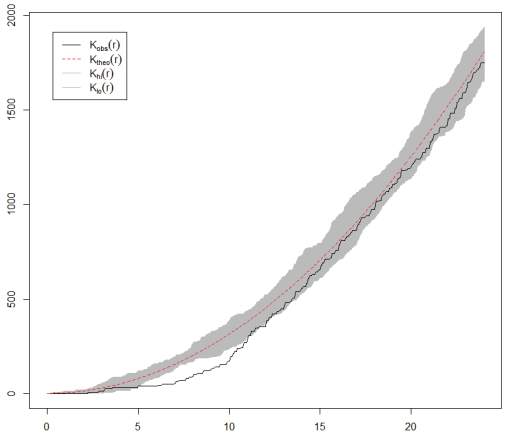
$$f_{min}(h) \leq \hat{f}(h) \leq f_{max}(h)$$

holds for all h, the model is accepted as a CSR, otherwise it is rejected.

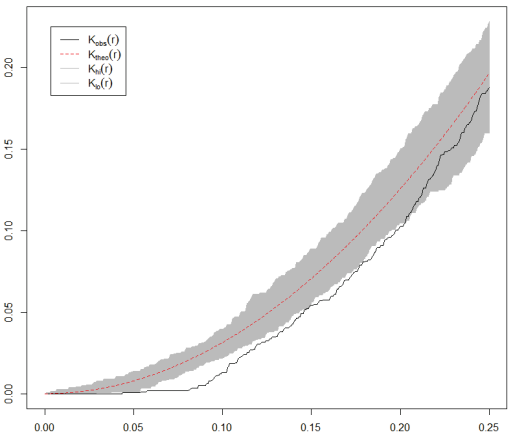
Figure 7.44 illustrates a sample of Monte Caro simulation, where the emperical curve of the $k - function$ is compared to a set of CSR simulation on order to check if the emperical curve lies within the CSR area or not.



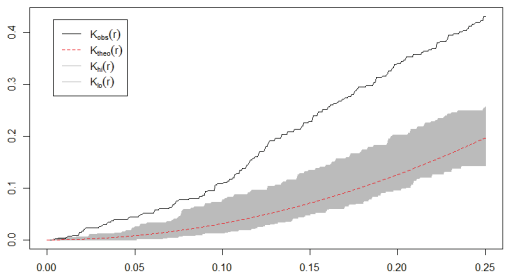
(a) clustered process



(b) Poisson process (homogeneous)



(c) regular process



(d) inhomogeneous Poisson process

Figure 7.44: Simulated examples of Monte Carlo envelopes based on Ripley's K-functions with significance level 0.05. The grey area shows the range of CSR simulations. The black curve indicates the empirical distribution which is examined whether the line lies within the grey area of Monte Carlo simulations

8. Material and Methods

> A technical solution may be defined as one that requires a change only in the techniques of the natural sciences, demanding little or nothing in the way of change in human values or ideas of morality.
>
> Garrett Hardin (In The Tragedy of the Commons, published in Science Journals, 1968)

We introduce the 3D-OSCOS (**3D-O**bject **S**egmentation and **Co**localization Analysis based on **S**patial statistics) algorithm which is implemented as a user-friendly toolbox for interactive detection of 3D objects and visualization of labeled images. It detects 3D objects in images with additional option for the user to interact with the program with the ability to delete or add objects. The basic idea of the toolbox is first to perform the object detection in a fully automatic way but in order to avoid a „black-box“ effect, the user has the opportunity to check the result visually and to correct it by deleting wrongly as object selected regions and by adding objects which are not detected by 3D-OSCOS.

8.1 Workflow of 3D-OSCOS

The well-developed 3D-OSCOS toolbox, as depicted in Figure 8.45, consists of five main steps. Each individual step of the this workflow may crucially affect the study result. This workflow starts with image acquisition and ends with a set of detected objects which is issued both visually and as a list of individual objects with their statistical properties. According to the proposed workflow in [Smal et al., 2010] where

small subcellular objects were detected, the 3D-OSCOS workflow (Figure 8.45) is organized as follows:

1. *Image acquisition* includes all steps from capturing images to forming a digital image data set.

2. *Image preprocessing* refers to all types of manipulation of acquired image, resulting in an optimized output of the image.

3. *Segmentation* includes all steps to subdivide an image in some connected meaningful regions referred to as objects. In general this step requires a priori knowledge on the nature and content of the images, which must be integrated into algorithms.

4. *Colocalization analysis* refers to the investigation of the spatial overlap between two or more subcellular objects of different image channels.

5. *Statistical analysis* sums up all descriptive and spatial statistic methods to measure and describe the properties of the segmented sub-objects.

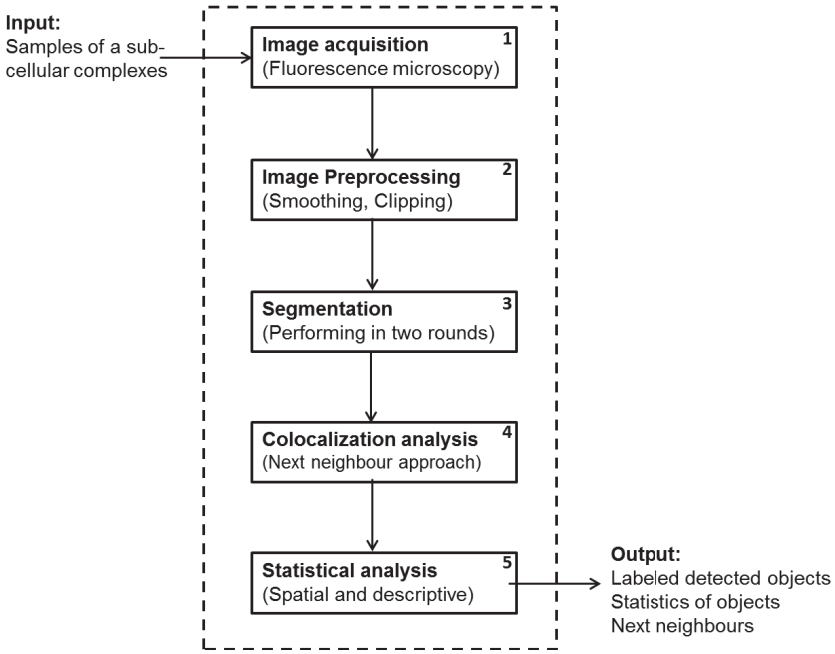


Figure 8.45: Workflow of 3D-OSCOS process. As it is illustrated, after sample preparation 3D-OSCOS performs 5 steps of image acquisition, image preprocessing, segmentation, colocalization and spatial analysis. The outputs are a set of labeled detected objects, their statistics and the next neighbor statistics.

8.1.1 Image Acquisition

Taking images with a digital camera requires training and experience, but this is just the first step to obtain a proper image. Image processing is used for establishing photometric accuracy so that pixel values in the image display the true values of light intensity.

As sketched in Figure 8.45, the process of colocalization analysis begins with the image acquisition. This step deals with the choice of the microscope and correct image acquisition settings. It should be bared in mind that some crucial instructions of image acquisition are essential for the whole pipeline. These points should be taken into account before the image is acquired by the microscope and afterthat post processing by computer. Understanding image acquisition techniques and their effects will help to design better experiments and improve the validity and

reproducibility of quantitative image analysis.

For the image acquisition step commonly a *charge-coupled device (CCD)* camera is used to acquire and record emitted light signal [Murphy, 2001]. The most essential parameters and effects of digital imaging are described in section 7.1.4. There are various terms that define imaging performance. These criteria can be categorized as follows:

- Spatial resolution (ability to capture fine details without seeing pixels)
- Signal-to-noise (clarity and visibility of object signals in the image)
- Numerical Aperture (the intensity of the signal captured by the microscope)
- Dynamic range (number of resolvable steps of light intensity, described as gray-level steps)
- Sampling (Fitting a single subresolution light source to an appropriate number of pixels on the detector to avoid over- or undersampling).

In order to test and evaluate the 3D-OSCOS workflow the *MSL2 3d-SIM staining* particles were imaged. MSL2 is a component of DCC (The dosage compentation complex), also known as the male-specific lethal (MSL). The image acquisition is performed with following settings: NA = 1.35, 100x objective plus additional 1.6x lens, 100 nm pixel size. We acquired z-stacks of 50 images each with a 400 nm z-spacing. Stacks were maximum projected prior to image analysis. For normal confocal imaging, voxel size less than or equal to 80 x 80 x 200 nm are usually perfectly acceptable. In our case we acquire images with a resolution of 39.4 x 39.4 x 120 nm.

8.1.2 Image Preprocessing

After visual inspection, it can be recognized that the desired 3D subcellular objects appear as bright spot-like peaks with a 3D-Gaussian form on a diffuse background light. Further due to limitations in imaging technology, a part of the spots are blurred and noisy. The brightness and contrast of the spots also varies and the illumination also differs over the slices.

The main goal of the preprocessing phase is to reconstruct true fluorescence spots and to suppress the background noise as good as possible based on image processing approaches like filtering and clipping [Ronneberger et al., 2008]. Regarding to the general image processing approaches (introduced in 7.1.4) and following different image investigations, smoothing filters and image clipping have proven to be appropriate in order to improve the image quality for a better statistical analysis. These two approaches are described as follows:

Smoothing filters

Among other filters, the *3D-Gaussian-filter* and *top-hat-filter* are applied in 3D-OSCOS workflow which belong to the *smoothing filters*. The image processing phase is concluded by using a clipping method to suppress all background pixels with intensity lower than a specific threshold. These three operations are described in the following sections:

Gaussian filter

The Gaussian filter is one of the most commonly used image processing filters. It improves images by reducing random noise that is usually caused by image acquisition due to accidental and nonspecific staining. In order to reduce the noise and to emphasize the spot centers, we apply 3D Gaussian filtering operation to make objects more homogeneous regarding to the intensity:

For the smoothing task a suitable 3D Gaussian kernel with a standard deviation (σ) equal to the desired spot size has to been defined as follows:

$$G(x, y, z) = \frac{1}{2\pi\sigma^2} \cdot e^{-\frac{x^2+y^2+z^2}{2\sigma^2}}$$

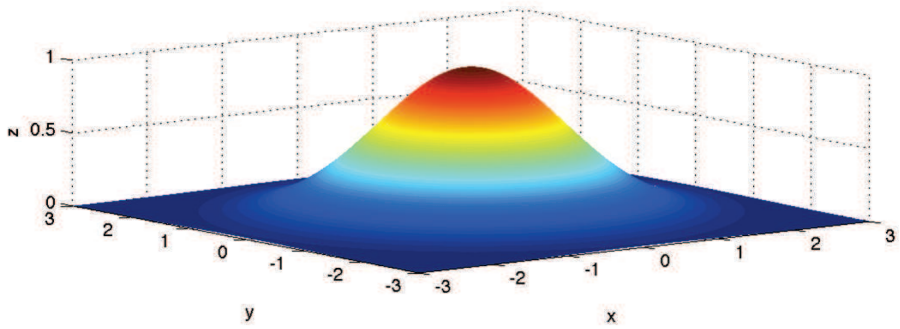


Figure 8.46: A sample illustration of 3D Gaussian bell curve in the range of -3 to 3.

The calculated Gaussian kernel can be convolved with the image. Gaussian filtering assigns each pixel a value average across the neighboring pixels within a certain radius. The improvement result is based on the assumption of the normal distribution of light intensity from a point light source and a random distribution of the additive noise. In general Gaussian filter may be recommended for images that suffer from random noise.

Top-Hat filtering

Due to varying intensities of spots, smoothing alone is not enough to distinguish the spots from the background. Therefore after denoising, in order to increase the contrast a top-hat filter [Gonzalez and Woods, 2008] is used.

Top-hat filter is a *morphological filter* (described in[Gonzalez and Woods, 2008]) and is used to enhance the signal and to correct uneven illumination of the image. It uses the *opening* operation from mathematical morphology. Morphologic operator is principally performed on binary input images. Similar to filtering, it uses a binary template, which is also referred to as *structural element*, is associated to the binary image using logical operations, such as *erosion* (based on logical AND), *dilation* (based on logical OR), *opening* (erosion followed by dilatation) and *closing* (dilation followed by erosion). An extensive description can be found in [Gonzalez and Woods, 2008].

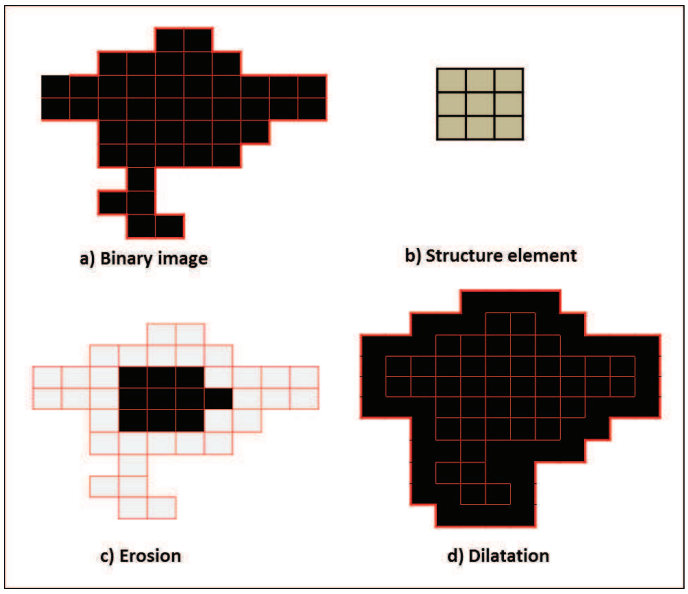


Figure 8.47: Binary morphology. a) The binary pattern is outlined in red. The resulted image area is in black. b) The defined structure element. c) and d) are the result applying the erosion and dilatation operation on the binary image, repectively.

The grayscale morphological top-hat filter acts as a local background removal function and at the same time enhances round, spot-like structures. Thus a grayscale

opening with a disk-shaped structuring element is performed and subtracted from the original image. An opened image is contained in the original image which, in turn, is contained in the closed image. As a consequence of this property, we could consider the subtraction of the opening from the input image, called top-hat.

More formally, the gray-scale 3D top-hat result is given as:

$$G_{\text{diff}} = I - \Gamma_r(I)$$

where $\Gamma_r(I)$ denotes the 3D opening operation using a disk shape structure element r which is of the desired object size.

In general the intra-slice resolution of images is higher than the inter-slice resolution, thus the size of the defined structural element spans more pixel in the x- and y-direction (intra-slice) than in the z-direction (inter-slice)

Image Clipping

It is well known that pixels deriving from noise should have lower intensities than pixels deriving from structures. In order to suppress the background and to emphasize the spots, an appropriate clipping threshold T_{Clipp} should be defined. All pixels with intensity below the clipping threshold are mapped to zero, otherwise the pixels remain unchanged. If $g(x, y, z)$ is a clipped version of voxel $f(x, y, z)$ at an appropriate clipping threshold T_{Clipp}, then:

$$g(x,y,z) = \begin{cases} f(x,y,z) & \text{, if } f(x,y,z) \geq \mathrm{T}_{Clipp} \\ 0 & \text{, if } f(x,y,z) < \mathrm{T}_{Clipp} \end{cases}$$

It should be noted that in order to specify an appropriate value for the clipping threshold, both the average and the contrast of voxel intensities should be considered, thus the clipping threshold T_{Clipp} can be defined as:

$$T_{Clipp} = \mu + c_1 \cdot \sigma$$

where μ and σ denote the mean and the standard deviation, respectively, and c_1 is a factor which is set by the user after visual screening of the image in the beginning of the segmentation process [Worz et al., 2010]. Figure 13.67 illustrates the effect of applying all three described image processing methods and additional binarization on a sample image.

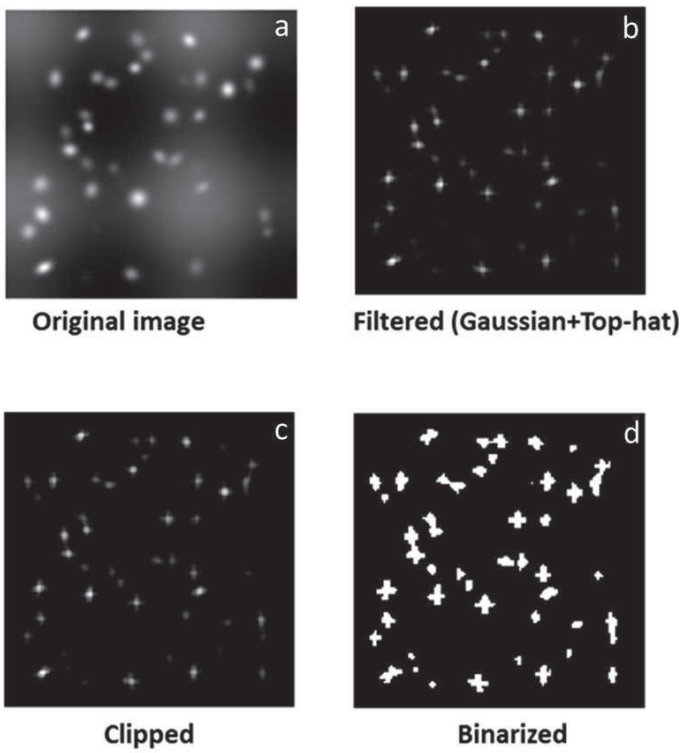


Figure 8.48: An illustration of image preprocessing results. A sample image layer (a) with the effect of filtering (b), clipping (c) and binarization (d).

8.1.3 Segmentation

Segmentation generally means dividing an image into connected regions corresponding to objects. The segmentation is usually based on identifying common properties.

Inspecting provided 3D Fluorescent images shows that two kinds of heterogeneities can be observed:

- Cell distribution heterogeneity: Various number of spots in different cell regions.
- Spot heterogeneity: Various spot volume size, spot average intensity and their pairwise distance.

We noted that both the inter-object distances and signal intensities differ in various regions of subcellular compartments. Therefore we suggest performing the segmentation phase in multiple rounds. First the objects are segmented and in a further step among all detected objects, those with a size greater than a defined threshold are analyzed separately in a further round to divide them into several smaller sub-objects fulfilling a user-defined range. Figure 8.49 shows a sketch that objects can be grouped into two types. The objects are either well separated or some of them are very closed-by and connected forming „big-objects“ which should be segmented in two separated segmentation routines, respectively.

First round: (3D connected component labeling)

To find all connected components, the topological relationship of adjacent pixels are analyzed. In other words, for each pixel the 26 voxels in the three dimensional neighborhood are inspected and all adjacent pixels above a certain threshold are considered to be a part of the same structure as the reference pixel. All pixels which are considered as a connected component are tagged or labeled with the same number.

To define an appropriate threshold, the well-known *global thresholding* [Otsu, 1979] is used which seeks to maximize between class variance [Gonzalez and Woods, 2008]. Due to the unequal intensity distributions among the slices, the global thresholding is applied for individual slice separately to get a threshold which is independent on the intensity distribution of other slices. The user has also the opportunity to affect the threshold by adding a factor which will be multiplied to the global threshold (called *threshold factor* c_2). Thus for each slice i, first the global threshold

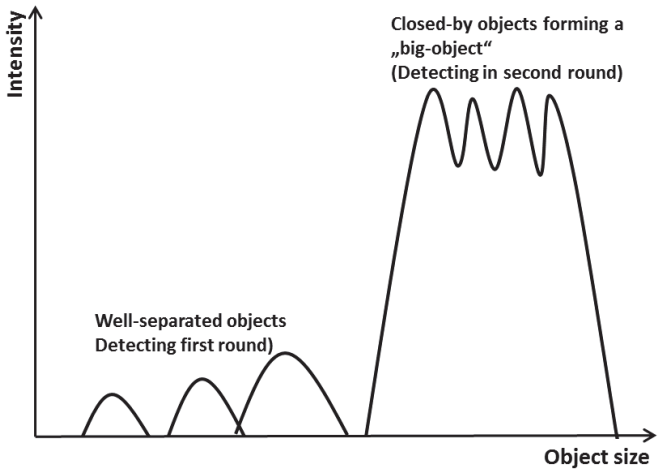


Figure 8.49: A sketch of well separated and close-by objects (forming a „big-objects“) based on the intensity and object size.

is determined and then multiplied by the factor c_2 which is defined by the user in advance:

$$T_{binarization} = \text{globalThreshold(slice(i))} * c_2$$

An appropriate threshold for each individual slice is determined by using a global thresholding approach and by considering the user defined a thresholding factor. All voxels with an intensity above the threshold will be considered as potential object‘s pixel other pixels will be considered as being part of the background. After binarization, all slices will be scanned from top left corner of first slice to the lower right corner of the last slice in order to find *3D connected components*. Each time a new object voxel with an intensity greater than zero is found, its 26 neighbors (9 neighbors on the upper and lower slice respectively plus 8 neighbors on the same slice) are checked for a voxel which doesn’t belong to the background that can be interpreted as its connected component. Each defined object (a set of connected components) is tagged by a unique number.

The following pseudo code outlines the first round of segmentation task implemented in MATLAB function „lableObjects.m“ as follows:

First round of segmentation:
Input: Threshold factor c_2, ObjSize = expected object size

1. For each slice i, calculate the binarization threshold $T_{binarization}$ as above.
2. For each pixel on each slice, check if $f(x, y, z) \geq T_{binarization}$, then map it to 1, otherwise to zero.
3. For each nonzero voxel, check the 26-neighborhood for connected components and define all voxels which are connected in 26 neighborhood as connected components.
4. Label each individual found connected component (refer to object) with an unique number
5. Determine for each detected object its properties, e.g. size, coordinates etc.
6. Divide all detected objects into two categories of objects:
   Normal objects: All objects with a size between ObjSize and 3×ObjSize
   Big objects: All objects with a size of greater than 3×ObjSize

Output: A list of all detected objects with their individual statistical properties.

Second round: (Local maximum search with distance)

The first round of segmentation depends on the volume size of the detected objects, following two categories of objects which can be defined:

- „Normal“objects
- „Big“ obejcts

where first category consists of objects which have a size in user-defined range. All objects of this group are added to the end list of detected objects. The big object category includes objects which are many time greater than the user defined expected object size (most of them are located normally in the X chromosomal space). The main task is to resolve the big objects in biological meaningful small sub-objects satisfying the defined object size range, assuming that each of them is derived from a single light emitting molecule.

The second round of segmentation deals with the task of dividing big objects into sub-objects, for this task a novel approach which is an extension of the well-known local maximum search by taking into account their inter-distance based on *Euclidean distance*, is introduced. First for each big object the number of desired sub-objects (N) should be determined depending on its size and the size of the expected objects. Afterthat all 3D local maximum points are searched, i.e those points which have the highest intensity within a local 3D window (26 neighbors). Then the algorithm looks for n brightest 3D local maximum points satisfying a minimum distance criterion. At the end of this step, each detected voxel fulfilling both conditions are acted as seed points and represent reference locations for the desired objects.

Based on the found seed points, all remaining voxels should be assigned to one of these points to form a sub-object. In order to specify each voxel which point is the most suitable seed point, the next neighbor approach is used and hence each remaining voxel is assigned to that seed point with the smallest Euclidean distance.

The following pseudo code outlines the second round of segmentation which performs dividing of big-objects in smaller sub-objects and is implemented in „analyseHuge-Objects.m“ and „findDistancedMaxima.m“ as follows:

Second of segmentation:
Input: List of big objects, ObjSize = expected object size, d = suitable object distance

1. For each big object (i) specify the desired number of sub-objects $N(i)$ as a function of big object size and user defined expected object size:
   N(i)=big object size(i)/ ObjSize.
2. Determine all 3D local maximum points $P_1 \dots P_n$ of object (i).
3. Find and identify the brightest point (P_1) of the big object.
4. Search for the next brightest point (P_2).
5. Check whether the Euclidean distance between P_1 and P_2 is greater than d or an Euclidean distance is greater than $d/2$ and at the same time it is on the same slice or not.
6. If the condition is fulfilled, add P_2 to the list of seed points, otherwise look for the next brightest 3D local maximum points $P_3 \dots P_n$ which fulfil the conditions.
7. Repeat steps til a number of N seed points are found.
8. For each remaining voxel, check based on the Nearest neighbor approach which seed point is the nearest point.
9. Assign each voxel to its nearest seed point to form individual sub-objects.
10. Label each individual found connected component (referr to object) with an unique number.
11. Determine for each detected object its properties, e.g. size, coordinates etc.

Output: A list of all detected sub-objects with their individual statistical properties.

8.1.4 Colocalization Analysis

As described in last sections, the output of the segmentation phase is a set of well separated 3D objects. In order to measure the colocalization degree between two channels, each detected object should be presented by its center, which is a 3D point. Thus, our approach for colocalization analysis is based on comparing the three-dimensional position of their respective centroids.

Colocalization analysis based on pairwise nearest neighbor:

It should be considered that the colocalization measurement based on nearest neighbor can be distinguished between colocalization of two objects from two channels or between all objects of both channels. First the colocalization between two objects checks for each point in one channel which point from other channel is its nearest neighbor. If the distance is lower than a specified threshold, both associated objects colocolize. In general two objects colocolize if their distance is below optical resolution [Boutte, 2006].

The pairwise distance and the nearest neighbor can be formally defined as:

$$\text{Pairwise distance: } d_{ij} = \|x_i - x_j\|, x_i \neq x_j$$
$$\text{Nearest neighbor distance: } nn_i = \min_{i \neq j} d_{ij}$$

Second, Lachmanovic et al. [Lachmanovich et al., 2003] introduces a measure of the colocalization degree between two channels as the proportion of the number of objects from first channel colocolizing with the objects from the second channel, to the total number of the objects from the first channel. As the number of objects in both channels may differ, the measurement has to be set to select objects from the channel with fewer objects and to search for the nearest neighbor from the channel with more objects.

$$\text{Degree of colocalization} = \frac{\text{number of colocalization}}{\text{total number of objects from first channel}}.$$

8.1.5 Statistical Analysis

A set of statistical properties of detected objects can be specified and summarized in a CSV file. This output data consists of all the detected objects row-by-row. Each object (presented as a row) is described by various statistical properties that are listed as columns.

Region properties of detected objects

After the object segmentation step, various statistical properties which provide hints of potential object interactions are of interest. Following features are specified by 3D-OSCOS toolbox:

1. Area:
This value reflects the actual number of pixels (or voxels in three-dimensional cases) in the object. In other words it describes the size in each individual detected object.

2. Centroid:
It specifies the center of mass of the objects.

3. Local maximum:
This parameter describes the value of the voxel with the highest intensity in the region. It additionally expresses its x-,y- and z- coordinates. This voxel serves also as a reference point for the object for colocalization analysis.

4. Descriptive statistics:
This category consists of main statistical features which is a summary of detected objects such as minimum, maximum, mean and standard deviation of all current object with voxels' intensities. Further the width, height and deepness of each detected object is reflected.

5. Number of objects:
Finally the number of objects is specially determined in three categories: Correctly detected objects (True Positives), wrongly detected objects (False Positives) and wrongly overlooked objects (False Negatives).

6. User defined settings and input image properties:
In order to enable the user to reproduce each executed routine, in the last worksheet all user input settings and input image properties are summarized .

8.2 Input and Output of 3D-OSCOS

The 3D-OSCOS toolbox requires an input image data and some user parameter settings which will be considered in the detection process. The input image data can be provided both as a set of multi-tiff files and as a single multi-stack file. This file format can be read directly by MATLAB. The only problem is that the image dimensions are not stored in TIFF data, whereas the toolbox cannot recognize what the correct distance between individual z slices is and how to translate the pixel size into micrometer. Therefore the user has to specify the dimension lengths of each voxel in x,y and z directions (e.g. x=0.0395 μm, y=0.0395 μm, z=0,12 μm).

8.2.1 User Interactions and Inputs

In order to obtain the best possible result, the user should determine in advance essential characteristics of desired subcellular objects through a preliminary investigation. These important facts should later be considered during the image and segmentation analysis. These facts are among others voxel size in x,y and z directions, expected object minimum, maximum size and average size, expected minimum and maximum object deepness and expected approximately object size.

Further program settings such as specifying the color of the channel (green or red), the path to the corresponding image data and the clipping and thresholding factors (described in section 8.1.2) are required. A program illustration of input windows and an extensive overview of input parameters are shown and described in Appendix II.

8.2.2 Program Outputs

One of the most exclusive and distinguishing feature of this toolbox is that the user can decide whether an inspection of the visual output is desired or not in order to verify its correctness and integrity. It offers the opportunity to perform a manual enhancement and correction of the output. In such cases, the user can add object labels or delete wrongly labeled objects from them by clicking on the appropriate location in the image layer. These two steps offer the opportunity to reduce *False Positive rate* (wrongly selected as object) and the *False Negative rate* (mistakenly overlooked objects).

The output of the toolbox can be grouped in three categories as follows:

Visual output of labels:

As shown in Figure 8.50, the visual output consists of three types of labels. Automatically labeled objects (depicted as $\odot$), manually post-labeling ($\boxdot$) and manually deleting of wrongly labeled regions ($*$). Furthermore, all three kind of labels are on both the real image background (Fig. 8.50 a and c) and on a black background (Fig. 8.50 b and d). The image background serves as an optical verification of the output and the black background is used for overlapping applications. Each labeled layer of the input image is saved in one file and all of them are stored in a single folder referred to as the input data set name.

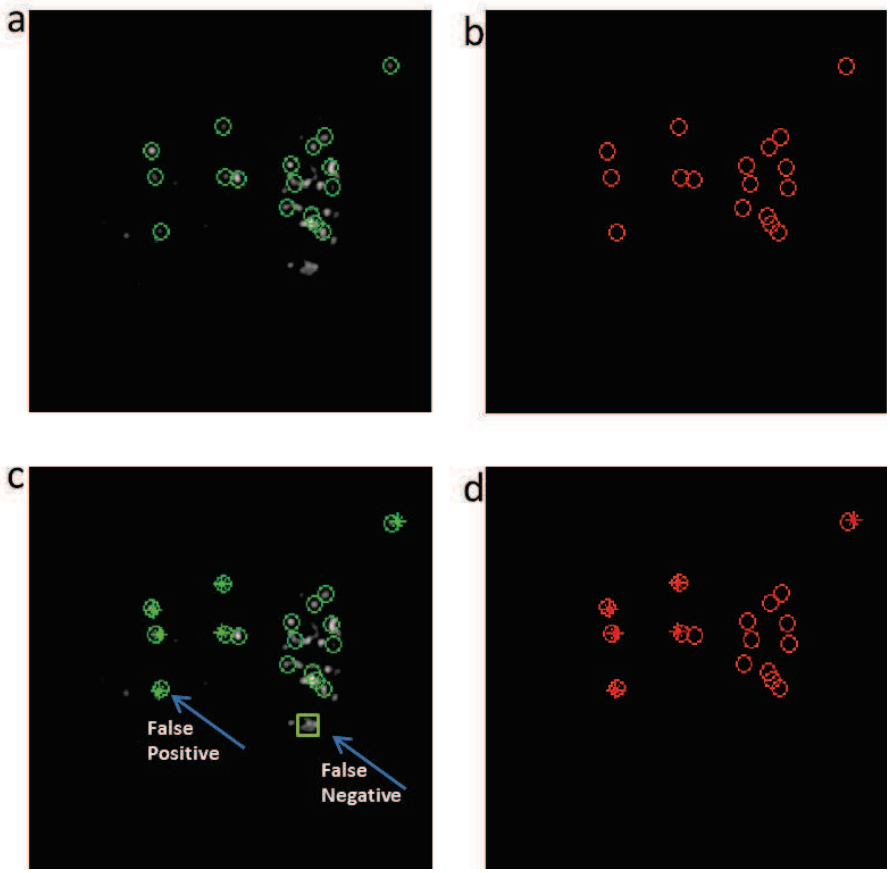


Figure 8.50: Visual output of labeled objects. Showing on a single layer of the 3D MSL2 protein image data set, where a) and b) show all automatically detected objects represented by a green circle on an original and black background respectively. c) and d) illustrates a combination of labels including manual post-labeling (False Negative) and manual deleting (False Positives).

Statistical properties of labeled objects:

The next output is a CSV file containing tables listing all detected objects with their properties, such as size, coordinates etc. Particularly the x, y, z coordinates are important inputs for the subsequent step of „Next neighbor distribution“ in order to find minimum distances between the objects. Further in this CSV file other essential parameters like user input setting values are saved, in order to make the result reproducible by knowing the input values. Moreover the list of all False Positives and False Negative is shown by manually clicking on the image during the post-labeling. Figures 8.51 and 8.51 show excerpts from these tables.

number	regionSize	Minimum	Mean	Maximum	standardDeviation	Width	Height	Deepness	CenterX	CenterY	CenterZ	spotCenterX	spotCenterY	spotCenterZ	ObjectNumber	datasetTyp	spotCenterX_mic	spotCenterY_mic	spotCenterZ_mic
1	11	1029	1507	2432	424	3	3	3	98	106	2	98	106	2	1	0	4,187	3,871	0,248
2	13	1043	1620	2450	428	3	3	3	103	105	5	103	106	5	8	0	4,187	4,0685	0,62
3	42	1047	2311	5957	1253	5	5	5	177	103	7	177	103	7	12	0	4,0685	6,9915	0,868
4	22	1053	1673	3357	607	5	4	3	70	97	8	70	97	8	15	0	3,8315	2,765	0,992
5	10	1209	1526	2157	335	3	2	3	66	105	9	66	105	9	20	0	4,1475	2,607	1,116
6	12	1143	1556	2216	354	2	3	3	177	91	9	177	91	9	26	0	3,5945	6,9915	1,116
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...

Figure 8.51: Output of statistical properties as a table showing all detected objects with their specified properties. The 3D coordinates of spot centers in both pixel and μm are displayed too.

MinRegSize	MinRegDepth	MaxRegDepth	FactClipping	MaxRegSize	FactGlobThresh	ObjectWidth	ObjectHeight	ObjectDepth
10	2	8	1	60	1	1	5	5

Figure 8.52: Output of user parameter settings. It shows all values which the user set in advance. This information offers the opportunity to reproduce the same result when running the algorithm again.

Next neighbor distribution between the objects:

An additional function for the analysis process is the next neighbor analysis (implemented in R Package: bioimagetools[5]). This function checks for each object represented by its coordinates the distance to the nearest neighbor. It can be performed either within a single channel (Fig. 8.54) which has no statistical significance for colocalization or between two different channels, which is a key measurement for colocalization analysis. Next neighbor distribution is a graphical representation (histogram) of the distances between each object from the first channel to its next

[5]https://r-forge.r-project.org/projects/bioimagetools/

neighbor from the second channel. It is a plot showing different distance scales between zero to maximum distance on the x-axis versus the number or relative number (frequency) of pairs with the appropriate distance for any given value of x on the y-axis. This toolbox can be used in the narrow sense to find out how many objects do colocalize. I.e. one can set a threshold on the x-axis as to define the left cumulative number of colocalized objects and on right side the number of objects that are far from each other. A very meaningful and common used reference value for threshold is the optical resolution of the image data (in our case $0.04\mu m$).

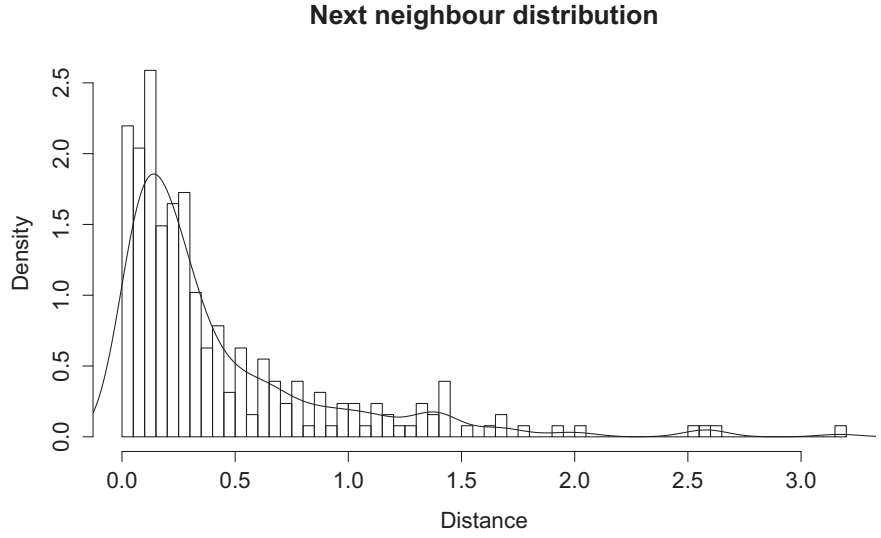


Figure 8.53: Histogram of distances between closed neighbours. From the objects lying on the red channel the distances to the objects of the green channel is considered.

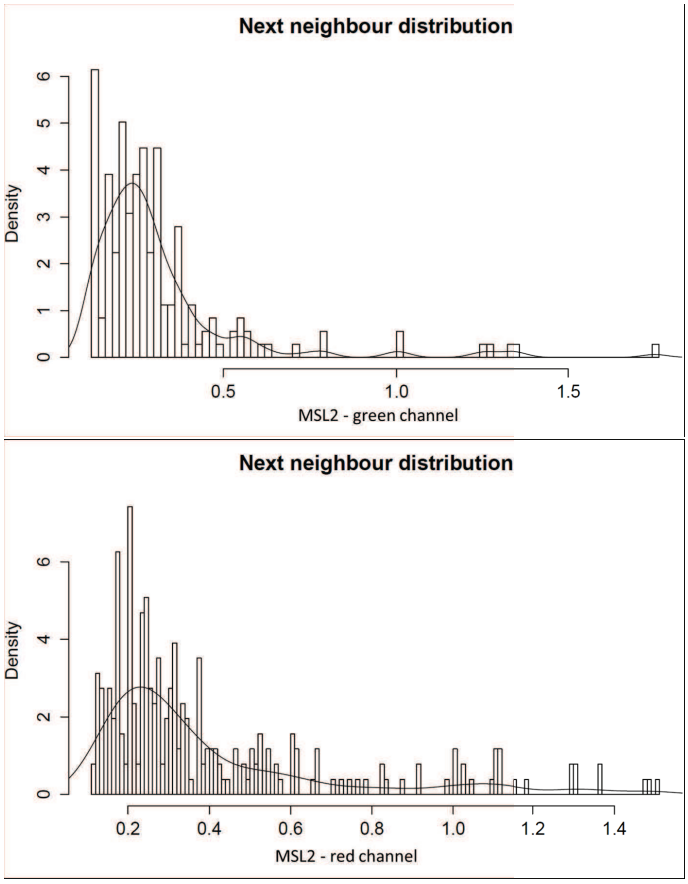


Figure 8.54: Next neighbor distribution of objects within a single channel.

9. Experiments and Results

> In questions of science, the authority of a thousand is not worth the humble reasoning of a single individual.
>
> Galileo Galilei

9.1 Validation of 3D-OSCOS

In order to quantify the performance of the 3D-OSCOS algorithm and to compare it against a ground truth reference, a set of simulated (computer-generated) images which consists of 3D images with different contrasts and signal-to-noise ratios was used. Further the introduced algorithm was compared to other algorithms proposed by [Bolte and Cordelieres, 2006], the ImageJ plugin *Object Counter3D* [Fabrice P. Cordelieres, 2006], and TANGO provided by [Ollion et al., 2013] in terms of accuracy and processing time. Moreover the 3D-OSCOS method was validated by applying an experimental data set and compared visually its output to the manual quantification result. Hence, the proposed method is evaluated based on both a real and an artificial data set.

For the performance measurement, the commonly used metrics proposed in [Fawcett, 2006] were applied as follows: First, a true positive (TP) is defined as a correctly founded object, and a false positive (FP) is a detected object for which there is no match in the reference image. A false negative (FN) corresponds to a missing object in the detection result. The same definitions may also be applied for pixel-level analysis described in [Ruusuvuori et al., 2010].

9.1.1 Validation based on Real Data set

Image analysis pipeline has been tested on foci recognition of *MSL2 3d-SIM* staining. *MSL2* (Male Specific Lethal 2 complex) is component of *DCC* (Dosage Compensation Complex) stains 5% sub-volume of nucleus of Drosophila male somatic interphase cells. For more description and details see [Georgiev et al., 2011].

The MSL2 complex can be roughly divided into three different regions which can be described as follows:

- Low intensity but well defined and well separated autosomal foci.
- High intensity, big size and well separated autosomal foci.
- Ill-separated close-by foci with varying signal intensities and sizes. These regions are the major challenge for the detecting task. The goal is to determine these big objects and divide them into several appropriate sub-objects.

The experimental microscope images have the disadvantage that first of all the ground truth is required. Creating a reliable and representative reference in biomedical applications is as mentioned a burdensome, difficult, time consuming and inaccurate task. Therefore, to enable comparisons against a reference result, simulated experiment benchmarks are used.

9.1.2 Validation based on Artificial Data set

In computer generated images the number and location of the spots are known, as they are simulated by computer with user-defined properties. Recently, benchmark image collections of cellular biological samples have been developed in order to facilitate comparison and validation of object detection and image analysis methods: [Ljosa et al., 2012a], [Ruusuvuori et al., 2010] and [Drelie Gelasca et al., 2009] [Ljosa et al., 2012b]. Unfortunately, the images are provided only as two-dimensional images, thus they are not suitable for our investigation. Therefore for the evaluation of our toolbox, we used a simulated 3D image developed by Wörz et al. [Worz et al., 2010].

The three-dimensional artificial image consists of exactly one hundred objects of a certain range in size which is distributed randomly over a 3D image. The image is provided in six different versions relating to the signal-to-noise relation. As depicted in Figure 9.55 the original image without any noise (Fig. 9.55 a) and

further images which are generated by adding different levels of random noise to the original image (Fig. 9.55 b-f). Thus all images consists of objects which are equally dispersed over them but with various signal-to-noise ratios. The provided three-dimensional artificial data set consists of 16 slices with a width and height of 128 pixels, respectively. The dynamic range of each voxel is 16 bit and the intensity of the voxels is very close together within a range of 9,960 to 10,040.

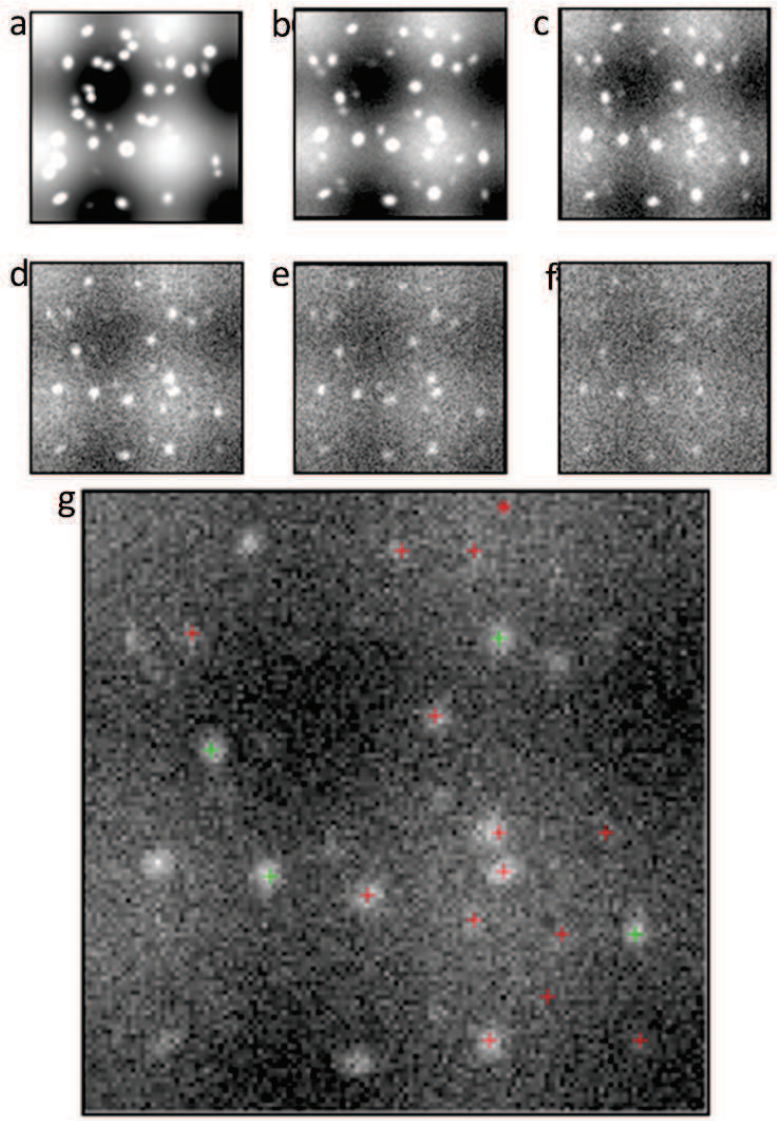


Figure 9.55: A sample slice of artificial image data. (a) Artificial image without noise, (b-f) Stepwise increasing of noise on the same image data. (g) The output of labeling process using 3D-OSCOS on the same slice.

9.2 Performance measurements

In order to measure and quantify the performance of the proposed method, a clear definition of ground truth is required. Thus in case of real data set, we define the manual detected objects as ground truth and in case of artificial data set, the number of objects is known since it is generated based on user inputs. Therefore we are able to compare our algorithm to other 3D object detection methods based on evaluation of the detection performance against a known number of ground truth.

9.2.1 Performance based on Artificial Data set

First we analyzed the performance, based on artificial data set. As it is shown in Table 9.3 the performance of 3D-OSCOS depends on the signal-to-noise ratio of each image type. The higher the noise, the lower the performance where the execution time for all image data sets is nearly the same. It should be mentioned that during the experiments some objects were indicated as double-labeled, which means that some objects that are expanded over several slices have become more than one mark. Therefore the final task was to remove all double-labeled by checking for unique number of marks in each object. It was performed based on Euclidean distance between the marks for each unique object.

The proposed algorithm is evaluated regarding to the spot detection task. We compared the 3D-OSCOS to two other algorithms proposed by [Bolte and Cordelieres, 2006] and [Ollion et al., 2013]. Ollion et al. have recently developed a generic tool for high-throughput 3D image analysis for studying nuclear organization TANGO (Tools for Analysis of Nuclear Genome Organization). It should be mentioned that the rest of the methods are either not suitable for 3D-images or they are not free available. The number of correctly detected objects using 3D-OSCOS is much closer to the real number compared to the Object counter plugin (see Figure 9.56).

Data set	Number of labeled objects	True Positive rate	False Positive rate	False Negative rate	Calculation time[6]
N00	109	99%	9%	1%	3s
N03	109	97%	9%	3%	3s
N10	112	97%	15%	3%	3s
N20	104	96%	5%	4%	3s
N30	95	91%	7%	9%	3s
N50	117	80%	15%	20%	5s

Table 9.3: The performance of 3D-OSCOS using artificial image. The number of detected object for each type of artificial image is listed. Based on these values the performance can be determined by giving the TP-, FP- and FN-rates. Each computer-generated image consists of exactly 100 objects. For example for the first data set (N00), our tool detected 109 objects, 99 of those were correctly detected and one object was not detected. Therefore, the TP-rate equals $99/100 = 99\%$, the FN-rate is $1/100 = 1\%$ and finally the FP-rate is equal to $10/109 = 9\%$. In addition the calculation or the running time of the automatic object detection is given.

Another study based on artificial data set compares the detection power of all three above methods regarding to their False Positive rates. As it is known the measurement of all four metrics (TP, TN, FP, FN) provides the basis to compare different methods regarding their performance. In our object detection case, determining TN makes no sense since the complementary part of objects is the background region. Further the FN can be derived from TP when the total number of objects is known (Total number of objects = TP+FN). Therefore we analysed the TP- and FP-rates that are summed up in Table 9.4:

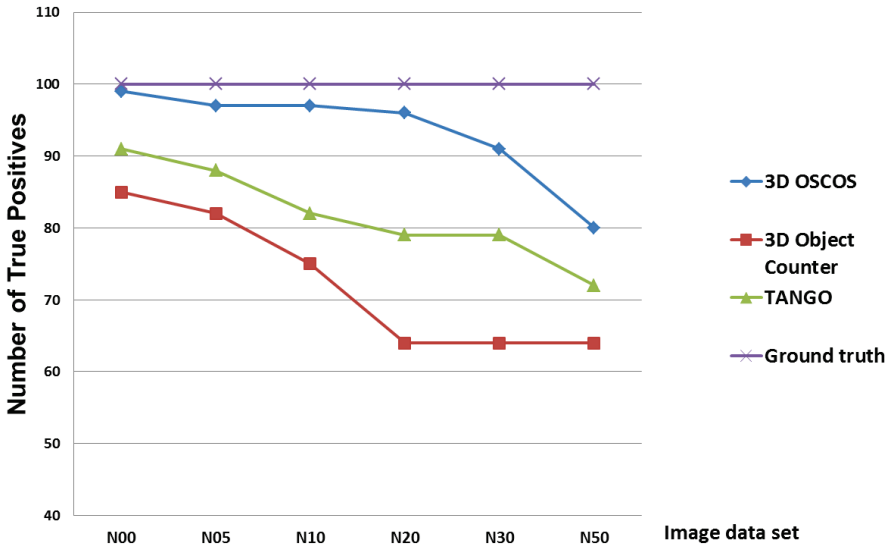
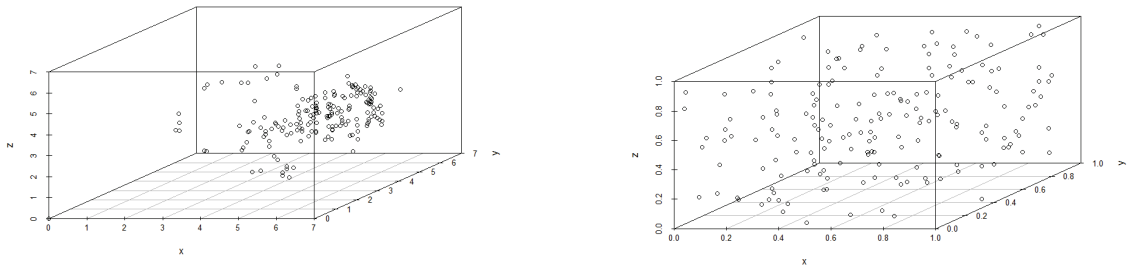


Figure 9.56: Performance of two different methods using artificial image data. The purple line shows the real number of objects in the image data (ground truth). The blue line relates to the 3D-OSCOS which is very close to the real number. The performance of the 3D object counter and TANGO plugin decrease more by increasing the noise ratio of the image data.

Data set	Proposed method		3D Objects Counter		TANGO	
	TP	FP	TP	FP	TP	FP
N00	99%	9%	85%	12%	91%	15%
N03	97%	9%	82%	11%	88%	16%
N10	97%	15%	75%	12%	82%	15%
N20	96%	5%	64%	15%	79%	18%
N30	91%	7%	64%	18%	79%	21%
N50	80%	15%	64%	24%	73%	23%

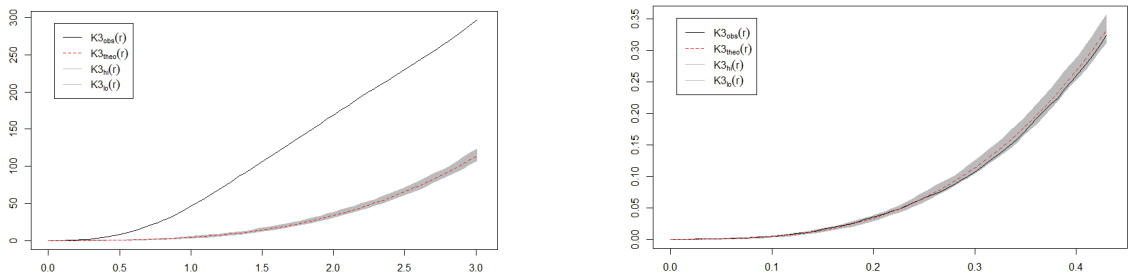
Table 9.4: The comparison of 3D OSCOS with two other methods provided by [Bolte and Cordelieres, 2006] and [Ollion et al., 2013]. In order to allow a better comparison, for each method applied on six different types of artificial image data set, both the True Positive rate and the False Positive rate are measured and compared.

As discussed in Section 7.4.1, spatial point processes provide a qualitative and quantitative characterisation of the object localizations represented by points. Figure 9.57 provides a visual inspection of the point distribution in a 3D space. It shows qualitatively, that the point distribution deviates clearly from the randomly distribution point patterns (CSR). Figure 9.58 confirms the vague initial impression from the visual inspection based on the Monte-Carlo test with a simulation of 99 $K(r)$ functions for Poisson process, where as shown the observed curve deviates from CSR. Hence, there is strong tendency for clustering for all values of r.



(a) observed process (b) Poisson process (homogeneous)

Figure 9.57: Observed 3d point distribution of artificial data set against a simulation of 3D randomly distributed point process.



(a) observed process (b) Poisson process (homogeneous)

Figure 9.58: An illustration of Monte-Carlo test based on a simulation of 99 K-function of 3D point processes simulated examples of Monte-Carlo envelopes based on Ripley's K-functions with significance level 0.05. The grey area shows the range of CSR simulations. The black curve indicates the empirical distribution which is examined whether the line lies within the grey area of Monte-Carlo simulations

9.2.2 Performance based on Real Data set

Concluding investigation of 3D-OSCOS is the analysis of the detection power of 3D OSCOS using the real image data set. We used the described *MSL2-3d-SIM* staining image data set. The result of manual detection is considered as reference or ground truth.

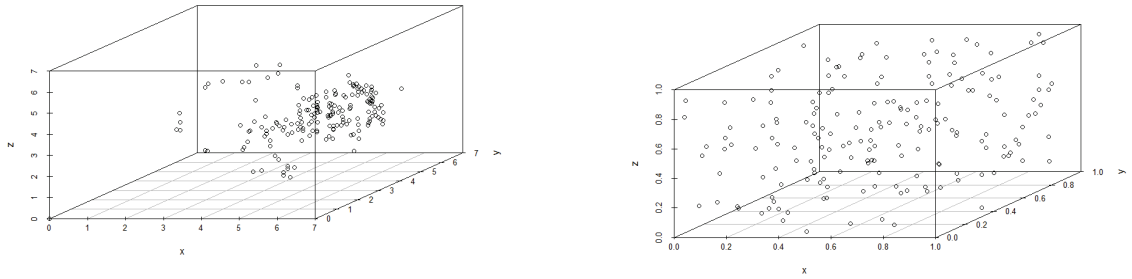
Image data	**TP**	**FN**	**FP**
MSL2-green channel	96%	4%	7%
MSL2-red channel	95%	5%	9%

Table 9.5: Validation of 3D OSCOS based on real image data set (MSL2 protein). Both channels are analyzed separately and the metrics TP-, FN- and FP-rates are determined in comparison to manual detection as defined ground truth. Based on manual inspection, we counted approx. 393 objects in green and 304 objects in the red channel, respectively. The performance rates are calculated based on these counted amounts. It should mentioned that the differences between the red and green channel is caused by image acquisition phase and unequal illumination factors.

The spatial point processes can be used in order to pursue the same goal to find out if the underlying distribution is completely randomly distributed (CSR) or not. For this investigation it is important to note that under the CSR assumption, the points are homogeneous Poisson distributed. For the first visual impression, Figure 9.59 shows qualitatively that the observed point process deviates from CSR.

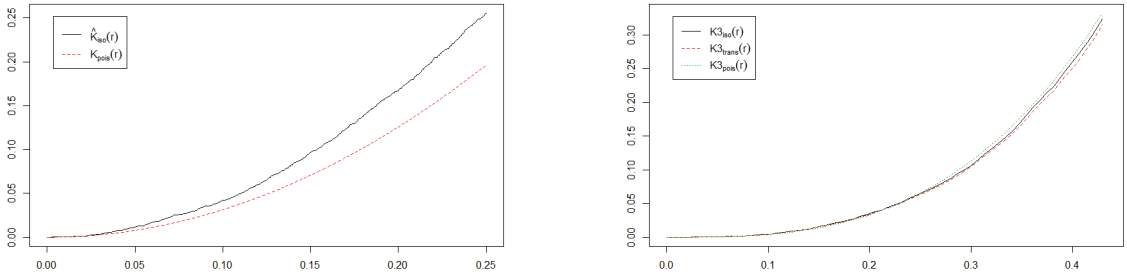
Next, the K-function approaches prove the CSR characteristic like first-order approaches, but based on the distance between the points and not their distribution. The shape of K-function tells us how the events are spaced in a point pattern, which confirms in this case again the deviation from the CSR (Figure 9.60)

Furthermore the statistical significance of assuming of CSR or departure from CSR can be evaluated using the Monte-Carlo approach. In cases where an arbitrary point pattern has to be investigated against CSR, a simulation of many randomly generated CSR curves (theoretical curves) is generated and the empirical curve of the point pattern is compared with this set of simulations. As depicted in Figure 9.61, the Monte-Carlo test based on the K-function shows no evidence against the assumption of an homogeneous Poisson process (CSR).



(a) observed process (b) Poisson process (homogeneous)

Figure 9.59: Observed localizations of 180 objects from the real 3D data set on left against a simulated Poisson point process on right.

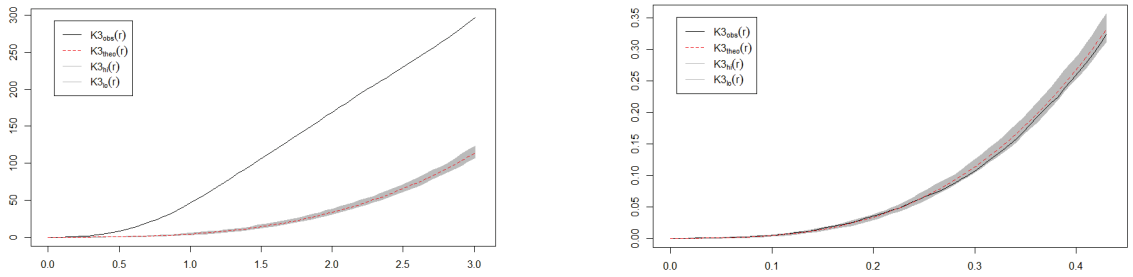


(a) observed process (b) Poisson process (homogeneous)

Figure 9.60: An illustration of K-function based on Observed real data set gainst simulation of a K-function based on 3D randomly distributed point process.

Last but not least the nearest neighbour distribution can be examined, which describes the frequency based on the distances from each point to its next neighbour. The distribution can be illustarted as a diagram, which provides a visual impression about the distances between objects in order to compare different point distributions or to find out if the most distances are less than the resolution, which indicates a colocalization between two channels (Figure 9.62).

In summary, while providing the ground truth based on real data set is very time consuming, since it requires a manual object detection to get the number of objects occurring in the 3D real image data, the artificial data set is more practical and can be performed fast and without much manual effort. On the other hand the real data



(a) observed process (b) Poisson process (homogeneous)

Figure 9.61: An illustration of Monte-Carlo test based on a simulation of 99 K-function of 3D point processes simulated examples of Monte-Carlo envelopes based on Ripley's K-functions of the real data set with significance level 0.05. The grey area shows the range of CSR simulations. The black curve indicates the empirical distribution which is examined whether the line lies within the grey area of Monte-Carlo simulations

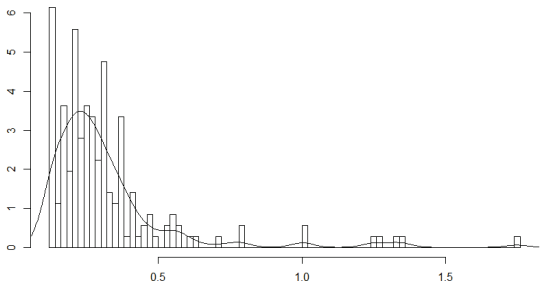


Figure 9.62: Next neighbour distribution between objects within a single channel.

set meets more the object properties and is more realistic. In case of producing of artificial image data, all properties and criteria should be specified and considered in order to achieve a more generalistic and significant results.

10. Conclusion of Part II

I never think of the future. It comes soon enough.

Albert Einstein (In interview given aboard the liner Belgenland, New York, December 1930)

Due to the huge amount of data being generated by the current generation of microscopes and the need of accurate statistical investigation, biologist lack of automated methods for quantitative analysis of cell nucleus and in particular for detecting of subcellular objects in microscope images. For these purposes, some problem-specific methods have been developed, where most of them only perform 2D analysis like Lerner et al. [Lerner B et al., 2007] and Raimondo et al. [Raimondo et al., 2005] (for an overview and evaluation review see [Ruusuvuori et al., 2010]).

There are some other tools which allow a 3D object detection, where they can be distinguished between freely available like IMAGEJ plugin JaCoP (described in [Bolte and Cordelieres, 2006]), NEMO Smart 3D-FISH detection tool introduced by [Iannuccelli et al., 2010]. Further, there are some commercial tools like IMARIS provided by *BITPLANE Scientific software* [Andor, 2013] (introduced by Costes et al. [Costes et al., 2004]) and *VOLOCITY 3D IMAGE ANALYSIS SOFTWARE* provided by PerkinElmer Group [Ram et al., 2010].

This part of the thesis is concerned with the detection of subcellular objects and the analysis their spatial proximity and possible interactions by considering the founded objects as a spatial point process. We have developed a fully automatic 3D object detection toolbox which provides a user-friendly interface in order to interact with the user for parameter setting inputs and to prevent a „black box“ effect. The 3D-OSCOS detects automatically objects in three-dimensional images

with an additional option for the user to check the result visually and if necessary to enhance it. An advantage feature of 3D-OSCOS is the option to run the program either fully automatic when the user has no prior information about the image or semi-automatic when the user wants to contribute his knowledge about the image data into the detection process. The statistical power and utility of this method is shown by comparing the detection power in comparison to other freely available methods.

Another notable finding of this work, according to the user intervention option during the detection phase, is that the inaccurate and time consuming task of manual object detection can be avoided as follows: The main part of the 3D detection phase can be executed automatically (e.g. 95% - 98% of the total workload) and for the last detailed optimization the user has the opportunity to inspect visually the result and enhance it when needed. Furthermore, the statistical significance of essential characteristics such complete randomness can be evaluated using appropriate methods from spatial point processes.

We established an extension of object detection and colocalization analysis in the sense that the statistical spatial analysis is used in order to analyse the full 3D spatial information about objects, their localizations and interactions. In order to achieve this goal, after detecting 3D objects and determining their statistical properties, each of them is presented by its center point in order to analyse their potential interactions based on well-developed statistical spatial approaches.

All findings and experiences in this field are freely available for the community as an open source MATLAB and R packages with further detailed documentations.

11. Discussion

> The only true wisdom is in knowing you know nothing.
>
> Socrates

There is a strong need in biological science for suitable cell imaging techniques in order to provide accurate analysis for getting insights within cells. Furthermore, due to the current progress being made in the area of image acquisition techniques, the amount of collected images has grown enormously with much more details. For these reasons manual analysis and basic approaches have reached their limits and the community is searching for innovative imaging and statistical approaches to face the challenges and provide appropriate and automatic approaches.

In this thesis, we have proposed and evaluated two methods of automatic particle selection and 3D subcellular detection, after giving some theoretical background and describing currently used methods. Both introduced methods (*MAPPOS* and *3D-OSCOS*) are provided as freely available and open source toolboxes for the community, which are conducted with additional algorithms and data sets. Further, both methods are evaluated on both experimental and artificial image data sets compared to other existing methods.

In more detail, the thesis roughly consists of two separate projects. The first project introduces *MAPPOS* an automatic particle picking toolbox integrated in the 3D Cryo-EM process based on image processing and Machine Learning techniques. The second project deals with the task of automatic or semi-automatic 3D object detection in fluorescence images which leads to the *3D-OSCOS* toolbox.

MAPPOS:

With regard to the huge amount of data being generated by the current generation of electron microscopes (the speed of data acquisition on a Titan Krios EM is 2.000 micrographs per day, which amounts to 200.000 particles per day), automated particle picking tools will inevitably become an integral part of every cryo-EM pipeline. Hand-picking of the E.coli data set requires many working days. We introduced MAPPOS, an ultra-fast particle picking method that reduces the amount of required manual preprocessing in the 3D Cryo-EM by orders of magnitude, while achieving a comparable specificity and sensitivity as for manual picking. MAPPOS is extremely fast, it can handle this deluge of data at the same pace at which it is generated. E.g., it took 1 hour for one person to generate the training set, and approx. 2 hours to run Mappos on a standard desktop computer. As we have demonstrated, the quality of the final 3D reconstruction equals to that of the hand-picked data set, although the quality of the raw data was moderate and required substantial filtering. The utility of Mappos is presumably maximal for unsymmetrical, large molecules, for which a large number of particles need to be picked for high resolution cryo Electron Microscopy.

One of our very essential, yet key findings of the first project, is that instead of improving of well-developed automatic particle picking methods, we have recommended performing the automatic routine of MAPPOS on cropped images after picking particles from micrograph to substitute the time-consuming and subjective task of manual post picking. The task of automatic particle picking from micrographs should be addressed separately from the task of automatic particle post picking perfomed on cropped images. MAPPOS has been successfully used to reconstruct a cryo-EM analysis of ribosome recycling complexes ([Becker et al., 2012] and [Leidig et al., 2012]).

3D-OSCOS:

During the second project, the 3D-OSCOS toolbox was developed in order to perform an automatic and precise 3D object detection especially in fluorescence images. 3D-OSCOS is a coherent framework allowing biologists to perform the complete analysis process of 3D fluorescence images by combining two environments: MATLAB for image processing, object detection and R for statistical analysis of measurement results. It interacts with the user in order to consider user parameter settings. The input data of 3D-OSCOS can be either a multichannel image or a set of image stacks.

A particular feature of this toolbox is the option that the user can decide whether to drive the experiments automatically or semi-automatic through user interaction. In order to improve the result of automatic object detection process a manual intervention is possible. Furthermore, essential prior information about the objects can be integrated into the analysis by user defined parameters.

Due to the high demand of reliable and fast measurements of the relative positioning of subcellular objects toward their chromosomal territories and their potential interactions, we have extended the 3D object detection by analyzing their positions in order to find any indication for direct or indirect interaction between subcellular structures. The spatial point process provide suitable approaches in order to find significant conclusions about the object localizations and their cellular functions.

Despite well-developed approaches for spatial point analysis and the widespread use in some fields such forestry and ecology, there are still some limitations for the employment in cell biology, whereas the most important restriction is that most of the introduces approaches are provided for just 2D data sets. Recently some investigators have started to adapt and extend the developed function for 3D purposes. However, as the experiments have shown, the spatial point analysis offers great potentials for analysing the nuclear localizations and their interaction in a statistically meaningful way by indicating the statistical significance and further essential parameters. In order to exploit the possibilities of spatial point modeling, the available methods should be expanded by appropriate 3D functions.

Appendix I & II

Appendix I

Implementation details

MAPPOS is written in MATLAB (version R2010). MAPPOS requires the Statistics Toolbox and Image Processing Toolbox as additional MATLAB packages. http://www.treschgroup.de/mappos.html

Inputs and outputs of MAPPOS

MAPPOS requires two sets of images. One set for learning which consists of sample boxed images (particles and non-particles). Each boxed image is saved with a unique filename correspondent to the image number and is saved in SPIDER format (e.g. „70s“). Please note that the sample images including particles and non-particle for learning phase should be provide in two different folders with the names „good“ for particles and „bad“ for non-particles. A further folder should be provided including all images, which should be classified by MAPPOS.

All images of both sets (learning and detecting sets) can have any pixel size, which will be specified by user at the beginning. They can be provided in all common formats (JPEG, TIFF, etc.) and also MAPPOS is able to read images in SPIDER format.

MAPPOS requires some parameter settings from the user. The output of MAPPOS is a text file, which is formatted in SPIDER syntax in order to be readable from SPIDER software without any manual modification. Therefore it is possible to continue the 3DEM reconstruction process in a full automatically manner.

As depicted in Figure 12.64, the text file consists of four columns: The first column is a continuous number of detected images, the second column is a value which indicates how many columns follow, the third column represents file numbers that correspond to the image numbers, which are classified as non-particles, and finally in the fourth column you find one which is an indicator for the micrograph type. Please note, that all character spacing between all values in each row (e.g. one or four spaces) are crucial and important according to SPIDER syntax.

As mentioned, MAPPOS replaces the manual post picking step which has the task to detect as many as possible non-particles to delete them from the data set. Therefore MAPPOS provides a list of all images that are classified as non-particles and a list of the counterpart (particles) is not relevant for further steps. The output file will be saved in the same folder, where the images for classifying are saved too.

Step-by-step Instruction

MAPPOS includes several steps from reading SPIDER images, feature extraction, learning a classifier, detecting non-particles to outputting a result file.

We describe concisely the major functions and routines of MAPPOS:

- **Start.m**:
Main function of MAPPOS in order to run it.

- **readsamplespiderimage.m**:
It moves to the folder with sample image and open both subfolders „good“ and „bad“ to read them. It reads all images depending on their formats and extensions. It saves both groups of images in two arrays „images_P“ and „images_NP“.
INPUT: Path to the folder with sample images are given by the user.
OUTPUT: Two arrays containing particle- and non-particle images.

- **allfeatureextraction.m**:
The input of this function is both arrays of particles and non-particles determining in the previous step. It extracts for each image the appropriate features mentioned in section 3.1.2. For each image and each feature it determines one feature value. It summarizes all results in one single table (data set matrix), where each row is presenting one image and columns indicate the determined features.

INPUT: Images of both groups (particles and non-particles).
OUTPUT: Data set matrix including all features.

- **generateclassifierensemble.m**:
Based on the data set matrix, including all features of both groups, the learning routines start by generating different classifiers and building an ensemble of individual classifiers described in section 3.1.3.
INPUT: Data set matrix with all features.
OUTPUT: Classifier ensemble.

- **allfeatureextraction.m**:
It extracts the same specified features as before from a new set of images that should be classified.
INPUT: Images which have to be classified.
OUTPUT: Data set matrix including all features.

- **calc.m**:
It applies the classifier ensemble on the data set matrix to get class labels (particle or non-particle).
INPUT: classifier ensemble, data set matrix.
OUTPUT: class labels 1 and 2 (1:particle or 2:non-particle)

- **numberofbadspideroutput.m**:
It checks all the images which are labeled as non-particle. Bring them to a SPIDER syntax (section 1.1) and saves them as a text file in the same folder where the images for classification are saved.
INPUT: List of labels
OUTPUT: A text file in SPIDER format containing all non-particle image numbers.

Parameter settings

Imaging and acquisition parameters for cryo-EM Data

E. coli 70S ribosomes were prepared as previously described in [Burma et al., 1985]. In short, a crude ribosomal fraction was further purified by using a liner 10-40% sucrose gradient. The monosomal fraction was then applied to $2nm$ pre-coated Quantifoil R3/3 holey carbon supported grids and vitrified using a Vitrobot Mark IV (FEI Company) and visualized on a Titan Krios TEM (FEI Company) microscope at 300 kV at a nominal magnification of 75,000 with a nominal defocus between 1µm and 3.5µm using an Eagle $4k \times 4k$ CCD camera (FEI Company, $4,096 \times 4,096$ pixel, 15µm pixel, $5s$/full frame) in a negative defocus range of 1.03.5µm. The resulting pixel size was 1.17 Åon the object scale.

Data was collected using the semi-automated software EM-TOOLS (TVIPS GmbH). This allowed the manual selection of the appropriate grid meshes and holes in the holey carbon film. During acquisition, the software automatically performed a re-centering, drift and focus correction before the final spot scan series were taken. Long-term TEM instabilities in beam shift, astigmatism and coma were corrected by EM-TOOLS in regular intervals (for example, every 45min).

Besides their classification performance, we assessed the effect of post-picking on the reconstruction quality of the electron density map. We defined an input data set consisting of 85,726 windowed projection images which were detected by the template matching algorithm of SIGNATURE. For automated classification, a training dataset of 2,000 particles (50% particles respective non-particles) was provided. All data sets were processed using SPIDER and refined for 3 rounds to a final resolution of about a Fourier-Shell Correlation (FSC0.5) of 11 Å.

Feature extraction parameters

Beside the mentioned parameters, there are some critical parameters, which are set after testing and analyzing many simulations with different combination of parameter values. Attention should be paid, among others, to the following imaging and feature extraction parameter:

- **Binarization threshold:** For binarization the threshold should be set. Global thresholding [Otsu, 1979] offers a good way to calculate the threshold.

- **Dark Dot Dispersion:** To find the points with a high intensity the range of the threshold should be in range of 95% quantile.

- **Canny edge detection:** This function returns a binary image of the same size, where all founded edge are assigns to one and the rest to zero. The Canny method finds edges by looking for local maxima of the gradient. The parameters for the threshold in our case are suitable in range of 0.35 to 0.45.

Sample run of MAPPOS

To start MAPPOS, the user has to start the main function by typing „start“ on the MATLAB environment. After that, the program expects following inputs from the user:

- **Image size:** Width and height of each boxed image in pixels (e.g. 368x368).

- **Resizing factor:** Decimation factor in image reading process. MAPPOS is able to read images directly in SPIDER format and to convert them into a grayscale image. The grayscale can be resized by user defined decimation factor (e.g. df = 0.2).

- **Paths of images:** Two paths of directories are requested. One directory includes all sample images for the learning phase and the other directory consists of a set of new images which should be classified.

- **Micrograph extension:** The extension of the micrograph (e.g. „.70s“).

After starting MAPPOS and user inputs for the above parameters, the MATLAB commands is as follow (see Figure 12.63):

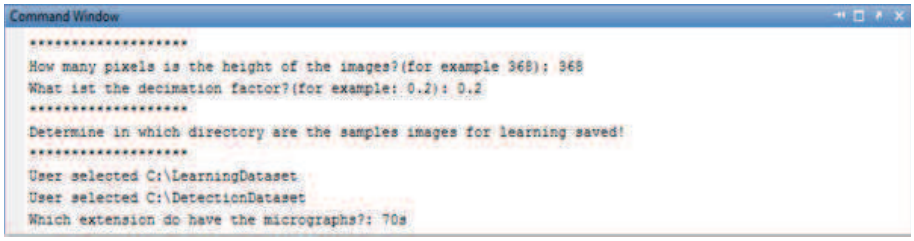


Figure 12.63: MATLAB command after using inputs. The user must first enter the size of each boxed image in Pixel and the decimation factor. Furthermore, MAPPOS asked for the paths, in which the dataset for training and testing are saved. Finally the user enters the extension of the micrograph (e.g. „70s“).

After reading all sample images, generating of classifier ensemble, MAPPOS classifies all images from the folder which are specified by the user. MAPPOS classifies all images into two groups: particle and non-particle images. For the reconstruction process we have to remove all non-particle images from the dataset therefore MAPPOS outputs a list of all non-particle image numbers. It saves a text file with the following content:

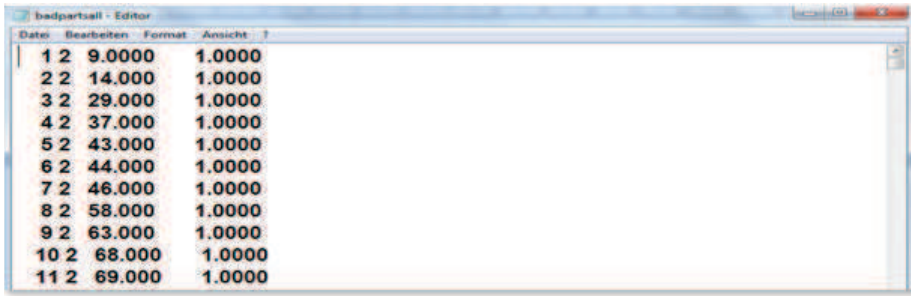


Figure 12.64: A part of the output text file in SPIDER format. The CSV-data consists of four columns. The numbers of the images which are classified as non-particle are listed in the third column. Therefore the SPIDER can delete all images which are listed in this data set

In the third column all image numbers which are classified as non-particle are listed. SPIDER software can read this text file and deletes all non-particle images. The post processing is done and the reconstruction process can be continued.

Differences between MAPPOS and others

EMAN2 (E2boxer) uses only positives to find other positives. It is difficult to set correct cutoff. In the publication, it is mentioned that EMAN2 needs manual polishing. Our goal is to replace the human expert in exactly this step.

SPIDER and IMAGIC are by far the most commonly used tools. SPIDER uses cross-correlation to filter for similarity of a boxed particle to a set of templates. This bears the problem that images with strong contrasts (e.g. „lines“ artifacts in our simulation) still yield high cross-correlations and tend to be overlooked. (For completeness, we included a SPIDER-postprocessed data set to show that this is insufficient).

All methods known to us require massive human intervention, which makes them inaccessible to objective evaluation criteria. We however have extended our validation framework substantially, and we demonstrate that MAPPOS truly performs as good as human experts.

The groups that we know do manual picking, or they use SIGNATURE software which picks only the particles. Conformational differences can only be taken into account for a reconstruction when sufficient quality is already available. Hence MAPPOS (or a manual- semi-automatic method) is a necessary prerequisite for assessing conformational changes.

Time consumption of MAPPOS

MAPPOS consists of two phases, the training and the detection phase. In the training phase a data set containing typically 1,000 particle images and 1,000 non-particle images is analyzed within approximately 73-75 seconds. The time is approximately linear in the number of images that are used for the training. In the detection phase, the classification of a single image takes roughly 0.3 seconds. Thus MAPPOS is able to classify about 12,000 images per hour. These figures were obtained on a standard PC with an Intel Core Duo CPU (2.00 GHz) and 2 GB RAM.

Appendix II

Requirements for running the program

The 3D-OSCOS algorithm is divided into two stages: Object detection and spatial analysis. The first step is implemented in MATLAB. It requires MATLAB version *R2009a* or newer one with an additional Image Processing Toolbox. The second step of the analysis is the spatial statistical analysis, which is implemented in R (the *bioimagetools* package[7]) based on *spatstat* package [Baddeley and Turner R., 2005]. The function *nndist.r* is the main function for analyzing and plotting of next neighbor distribution of 3D point patterns. It computes for each detected object-center, the minimal distances to the next neighbor and plots the distribution over all determined next neighbors distances.

Main functions of 3D-OSCOS

As mentioned in 8.1 the 3D-OSCOS consists of five main steps of image acquisition, image preprocessing, segmentation, colocalization analysis and statistical analysis. These five main steps are implemented in MATLAB based on following functions:

1. Image Denoising and smoothing:
(MATLAB files: ImageSmoothing.m, imfilter.m)
INPUT: Image stacks, expected object size
OUTPUT: Filtered image by performing Gaussian - and Top hat filters on each slice of the image.

[7]http://r-forge.r-project.org/projects/bioimagetools/

2. Image clipping
(MATLAB files: ClippIm.m)
INPUT: Filtered image, user defined clipping threshold factor
OUTPUT: Clipped images in stacks

3. Labeling connected components
(MATLAB files: labelObjects.m, im2bw.m, bwconncomp.m)
INPUT: clipped Image, user defined gray threshold
OUTPUT: List of labeled connected objects. First image binarization and then searching for connected components in binary image by analyzing the 26 neighborhood. The detected objects are distinguished between normal and mega objects.

4. Determining all object properties:
(MATLAB files: regionProperties.m)
INPUT: Labeled objects, mega objects and user defined parameter settings
OUTPUT: Properties of detected objects, considering the restrictions of expected objects. Object properties : Centroid, region size, Min intensity, etc.

5. Analysing mega objects
(MATLAB files: analyseSuperMegaObjects.m)
INPUT: Mega objects (all objects larger than a user defined size)
OUTPUT: Several sub-objects which are generated by dividing each mega object in several smaller objects.

6. Checking for closely objects
(MATLAB files: checkForClosedObjects.m)
INPUT: All object centers
OUTPUT: All object centers satisfying a minimum distance criteria

7. Presentation of the detected objects
(MATLAB files: outputResult.m)
As mentioned, there are two kinds of outputs: xls.file consisting of a list of all detected objects with their statistical properties and a visual output showing all image slices with detected objects presented by their centers.

8. Manual post labeling of objects
(MATLAB files: labelpoints.m)
The user has the opportunity to correct the output of the 3D-OSCOS process by deleting wrongly detected regions and adding some missing objects.

Parameter settings

As described above, the user can set some parameter values in prior to the analysis, which should be taken into account during the object detection phase. The list of these parameters are as follows:

- Voxel size in μm (x,y,z)
- Expected object size
- Minimum and maximum expected object size
- Minimum and maximum expected object depth
- Factor for clipping c_1 ($T_{clipp} = \mu + c \cdot \sigma$)
- Factor for global thresholdingn c_2
- Factor for brightest point of region

It should be mentioned that the tool provides additionally the opportunity to analyze the image in prior to the detection process in order to have an insight into the object properties (e.g. intensity, size, etc.). Thus the user can get appropriate information about the image and its occurred objects in the preprocessing phase. The *imageJ* tool can be also used for this purpose.

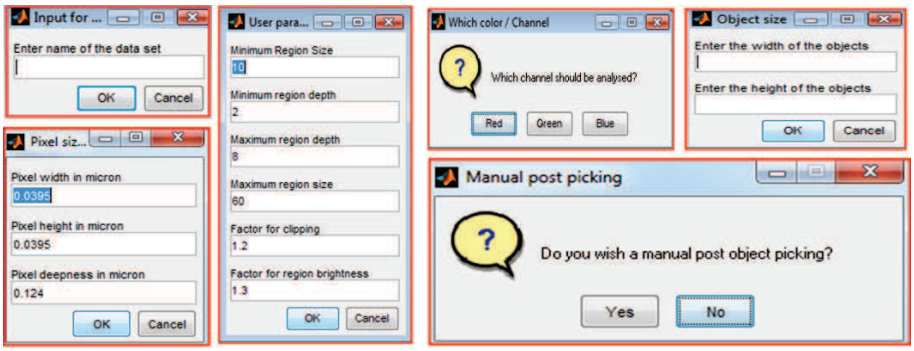


Figure 13.65: Screen shot of 3D-OSCOS parameter setting step. In each step one parameter is asked for and some default values are given for cases when the user has no idea.

ImageJ

ImageJ is an extensible, Java-based image processing application that is mainly dedicated to 2D images, but also supports image stacks. ImageJ supports essentially all standard image processing functions such as convolution, smoothing, filtering, edge detection, sharpening, morphological operators, etc. ImageJ can be extended by recordable macros and JAVA plug-ins. More than 500 plug-ins are available from the ImageJ website[8], which differ significantly in performance, quality and usability.

The output of the program

All output types (xls file and visual output) of the 3D-OSCOS are saved in a single folder according to the name of the dataset, which the user specifies at the beginning. The x-,y- and z- coordinates of foci centers converted in micron are listed in the columns 18, 19 and 20 of the xls file.

[8]http://rsbweb.nih.gov/ij/

Preprocessing and analysis of image data

In some cases it occurs that the user has no prior information or idea about the desired objects. For such cases the implemented tool provides the opportunity to perform different simulations based on various parameter combinations. For each parameter setting the user can specify the range of values that should be analyzed. Each parameter combination is executed and the associated number of detected objects is specified. In the end, as depicted in Figure 13.66, a diagram indicating the number of detected objects on the y-axis versus simulation number on the x-axis is represented.

Based on this simulation result, the user can find out the range of appropriate values for each simulation parameter. After the preprocessing step, the object detection phase can be started.

number	number of objects	MinRegionSize	minDeepness	maxDeepness	hugeObjectSize	clippingThreshold	brightesFactor	ObjectWidth	connectivity
1	409	10	2	8	80	2,5	1,5	5	6
2	401	10	2	8	80	2,5	1,5	5	18
3	395	10	2	8	80	2,5	1,5	5	26
4	344	10	2	8	80	2,5	1,6	5	6
5	337	10	2	8	80	2,5	1,6	5	18
6	331	10	2	8	80	2,5	1,6	5	26
7	290	10	2	8	80	2,5	1,7	5	6
8	283	10	2	8	80	2,5	1,7	5	18
9	279	10	2	8	80	2,5	1,7	5	26
10	389	10	2	8	80	2,6	1,5	5	6
11	384	10	2	8	80	2,6	1,5	5	18
12	383	10	2	8	80	2,6	1,5	5	26
13	319	10	2	8	80	2,6	1,6	5	6
14	315	10	2	8	80	2,6	1,6	5	18

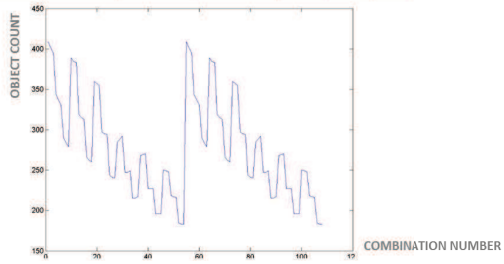


Figure 13.66: The output of different parameter simulations. The following are two outputs: A table shows for each simulation the number of detected objects with its associated specified input parameter settings. The diagram indicates the information in the first two columns of the above table.

Another possibilty in order to get a visual impression of different parameter settings, the 3D-OSCOS provides an illustration of each processing step based on a 2D stack (see Figure 13.67)

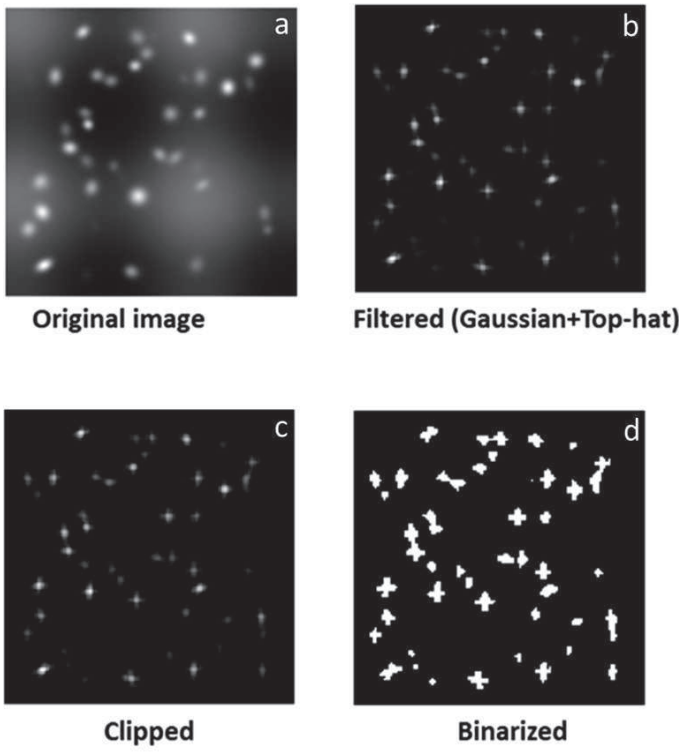


Figure 13.67: A sample illustration of visual parameter setting. The tool shows for each parameter combination a visual output. This figure shows the output for following parameter settings: masksize=7, clipping threshold = 0.1 and threshold factor = 0.9

Abbreviations

CCD; Is a small, centimeter-size chip of silicon that is divided up into millions of tiny picture elements which is able to store photoelectronic signals. The efficiency of light collection is so great that even weak images can be recorded in just a few milliseconds.

Centroid: The geometric centre of a nuclear compartment, when the compartment is nonspherical (e.g. chromosome territories), the centre is usually taken as the intensity gravity centre.

CLSM: Confocal Laser Scanning Microscope.

Colocalization: Two object types are regarded as „colocalzation“ if the NN distance between objects is (on average) 1)below some preset threshold, or 2) statistically significantly smaller than a population average NN distance. Two objects colocalize if they exhibit the (stochastic) tendency to lie in proximate spatial regions.

DCC: Dosage Compensation Complex.

FISH: Fluorescence in Situ Hybridization.

Fluorescence: A form of photoluminescence which persists only for a short period of time (usually less than 100 nanoseconds) after the cessation of excitation.

Foci: Plural of focus. The origin or centre of a disseminated disease.

Image segmentation: Images are often segmented in order to separate out regions of the image corresponding to objects from those regions corresponding to background. Segmentation is determined by a single parameter known as the threshold value. I.e. detection of nuclei and nuclear structures.

Monte Carlo test:
A way of assessing the computed statistical significance by random relabeling of the objects in order to recalculate distances (or other summary statistics) and thus to provide an empirical null distribution to calibrate the actual observed statistic.

Laser: (Acronym for Light Amplification by Stimulated Emission of Radiation.) A source which emits coherent radiation of high spectral concentration and an extremely small solid angle (low divergence).

Loci: is plural of locus, which means in genetics sense the location of a gene (or of a significant sequence) on a chromosome, as in genetic locus.

Magnification: The process of enlarging or the degree by which the dimensions in an image are (or appear to be) enlarged with respect to the corresponding dimensions in the object.

MSL2: Male Specific Lethal 2 complex.

Noise: „Random“ signal generated in the detection system (PMT and amplifiers) that is mixed with the real signal from the specimen. It is usually seen as a random pattern of speckles over the entire image. High gain values generate more noise in an image.

Numerical Aperture (NA): Numerical aperture is a measure of the acceptance angle of an objective and can also be considered as the ability of the objective to gather light and resolve specimen detail at a fixed objective distance. It is defined by the expression: NA = n(sin(a)), where NA is numerical aperture, n is the refractive index of the medium between the objective front lens and the specimen, and a is the half angle aperture of the objective.

Objective: The first part of the imaging system. It forms the primary image of the object.

Photon: Elementary quantity of radiant energy (quantum) whose value is equal to the product of Plancks constant h and the frequency (hz) of the electromagnetic radiation.

Pixel Dwell Time: The time spent illuminating and collecting signal from a single pixel position.

Pixel Size: The actual (real world) dimensions of a pixel, usually quoted as (x dimension, y dimension).

Pixels: Every pixel in an image has a pixel value, or intensity, indicating how bright that pixel is. For a grayscale image the pixel value is a single number with a range of possible values between zero and 255, where zero is black and 255 is white. Values between zero and 255 make up the different shades of gray.

Radial analysis/peeling: An approach to hypothesis testing that involves partitioning the nuclear volume into non-overlapping, contiguous regions. The test statistic compares the observed numbers of compartments in each region with the expected numbers under some null hypothesis, often complete spatial randomness.

Random (spatial) distribution: The location of a compartment is random if it cannot be precisely specified prior to observing the nucleus. Common use is taken to imply that any location is equally likely („uniformity“).

Resolution: The minimum distance separating two points such that these points can be seen as separate objects. Sufficient specimen contrast is also required to separate the two points.The minimum distance measurable between two points

SEM: Scanning Electron Microscopy.

SNR: Signal-to-noise ratio.

TEM: Transmission Electron Microscopy.

Thresholding: A specified pixel value is chosen as the threshold value. During thresholding, pixels in the image lower than the threshold value are set to zero (or black) and pixels higher than the threshold are set to 255 (or white).

Uniformity: The uniform distribution assigns the same probability to regions of equal size. In terms of the spatial distribution of compartments (with some technical assumptions), a uniform spatial distribution would be taken to mean complete spatial randomness.

Voxel: The three dimensional equivalent of a pixel. Thought to be derived from the term Volume Element. In other words, it is a volume element, the 3D analogue of a pixel

Wavelength: The distance on a periodic wave between two successive points at which the phase is the same. It is represented by the symbol (λ) and is usually expressed in nanometres and commonly refers to the colour of light

Bibliography

[Adiga et al., 2005] Adiga, U., Baxter W.T., Hall R.J., Rockel B., Rath B.K., Frank J., and Glaeser R. (2005). Particle picking by segmentation: a comparative study with spider-based manual particle picking. *J Struct Biol 152*, pages 211–220.

[Andor, 2013] Andor (2013). Imaris scientific 3d/4d image processing & analysis software bitplane.

[Andrey et al., 2010] Andrey, P., Kiêu, K., Kress, C., Lehmann, G., Tirichine, L., Liu, Z., Biot, E., Adenot, P.-G., Hue-Beauvais, C., Houba-Hérin, N., Duranthon, V., Devinoy, E., Beaujean, N., Gaudin, V., Maurin, Y., Debey, P., and Zimmer, C. (2010). Statistical analysis of 3d images detects regular spatial distributions of centromeres and chromocenters in animal and plant nuclei. *PLoS Computational Biology*, 6(7):e1000853.

[Arbelaez et al., 2011] Arbelaez, P., Han, B., Typke, D., Lim, J., Glaeser, M., and Malik, J. (2011). Experimental evaluation of support vector machine-based and correlation-based approaches to automatic particle selection. *J Struct Biol 175*, pages 319–328.

[Baddeley, 2006] Baddeley, A. (2006). *Case studies in spatial point process modeling*, volume 185 of *Lecture notes in statistics*. Springer, New York.

[Baddeley and Turner R., 2005] Baddeley, A. and Turner R. (2005). spatstat: An r package for analyzing spatial point patterns. *Journal of Statistical Software 12 (6)*, pages 1–42.

[Bailey and Gatrell, 1995] Bailey, T. C. and Gatrell, A. C. (1995). *Interactive spatial data analysis*. Longman Scientific & Technical and J. Wiley, Harlow Essex and England and New York and NY.

[Becker et al., 2012] Becker, T., Franckenberg S., Wickles S., Shoemaker C.J, Anger A.M, Armache J.P, Sieber H., Ungewickell C., Berninghausen O., and Daberkow

A.I. (2012). Structural basis of highly conserved ribosome recycling in eukaryotes and archaea. *Nature 482*, pages 501–506.

[Belmont et al., 1986] Belmont, A., Bignone, F., and Ts'O, P. (1986). The relative intranuclear positions of barr bodies in non-transformed human fibroblasts. *Experimental Cell Research*, 165(1):165–179.

[Besag, 1977] Besag, J. (1977). Discussion of dr ripley's paper. (193-195).

[Bishop, 2006] Bishop, C. M. (2006). *Pattern recognition and machine learning*. Information science and statistics. Springer, New York.

[Bolte and Cordelieres, 2006] Bolte, S. and Cordelieres, E. (2006). A guided tour into subcellular colocalization analysis in light microscopy. *Journal of Microscopy*, 2006(224).

[Boutte, 2006] Boutte, Y. (2006). The plasma membrane recycling pathway and cell polarity in plants: studies on pin proteins. *Journal of Cell Science*, 119(7):1255–1265.

[Bozzola and Russell, 1999] Bozzola, J. J. and Russell, L. D. (1999). *Electron microscopy: Principles and techniques for biologists*. Jones and Bartlett, Sudbury and Mass, 2 edition.

[Bradley, 1997] Bradley, A. (1997). The use of the area under the roc curve in the evaluation of machine learning algorithms. *Pattern Recognition 30*, pages 1145–1159.

[Breiman, 1993] Breiman, L. (1993). *Classification and regression trees*. Chapman & Hall, New York and N.Y.

[Breiman, 1996] Breiman, L. (1996). Bagging predictors. *Machine Learning 24*, pages 123–140.

[Burma et al., 1985] Burma, D., Srivastava A., Srivastava S., and Dash D. (1985). Interconversion of tight and loose couple 50 s ribosomes and translocation in protein synthesis. *Journal of Biological Chemistry 260*, pages 10517–10525.

[Canny, 1986] Canny, J. (1986). A computational approach to edge detection. *IEEE Trans Pattern Anal Mach Intell 8*, pages 679–698.

[Carpenter et al., 2006] Carpenter, A. E., Jones, T. R., Lamprecht, M. R., Clarke, C., Kang, I., Friman, O., Guertin, D. A., Chang, J., Lindquist, R. A., Moffat, J., Golland, P., and Sabatini, D. M. (2006). Cellprofiler: image analysis software for identifying and quantifying cell phenotypes. *Genome Biology*, 7(10):R100.

[Chen, 2007] Chen, J. N. G. (2007). Signature: a single-particle selection system for molecular electron microscopy. *J Struct Biol 157*, pages 168–173.

[Commenges, 2011] Commenges, D. (2011). Handbook of spatial statistics edited by gelfand, a. e., diggle, p. j., fuentes, m. and guttorp, p. *Biometrics*, 67(2):671–672.

[Costes et al., 2004] Costes, S. V., Daelemans, D., Cho, E. H., Dobbin, Z., Pavlakis, G., and Lockett, S. (2004). Automatic and quantitative measurement of protein-protein colocalization in live cells. *Biophysical Journal*, 86(6):3993–4003.

[Cover and Hart, 1967] Cover, T. and Hart, P. (1967). Nearest neighbor pattern classification. *IEEE Transactions on Information Theory*, 13(1):21–27.

[Cover and Hart P., 1967] Cover, T. and Hart P. (1967). Nearest neighbor pattern classification. *Information Theory, IEEE Transactions on 13*, pages 21–27.

[Cristianini and Shawe-Taylor, 2000] Cristianini, N. and Shawe-Taylor, J. (2000). *An introduction to support vector machines: And other kernel-based learning methods*. Cambridge University Press, Cambridge and New York.

[Daley and Vere-Jones, 1988] Daley, D. J. and Vere-Jones, D. (1988). *An introduction to the theory of point processes*. Springer series in statistics. Springer-Verlag, New York.

[Diggle, 2003] Diggle, P. (2003). *Statistical analysis of spatial point patterns*. Arnold and Distributed by Oxford University Press, London and New York, 2 edition.

[Drelie Gelasca et al., 2009] Drelie Gelasca, E., Obara, B., Fedorov, D., Kvilekval, K., and Manjunath, B. S. (2009). A biosegmentation benchmark for evaluation of bioimage analysis methods. *BMC Bioinformatics*, 10(1):368.

[Duda et al., 2001] Duda, R. O., Hart, P. E., and Stork, D. G. (2001). *Pattern classification*. Wiley, New York, 2 edition.

[Erni et al., 2009] Erni, R., Rossell, M., Kisielowski, C., and Dahmen, U. (2009). Atomic-resolution imaging with a sub-50-pm electron probe. *Physical Review Letters*, 102(9).

[Fabrice P. Cordelieres, 2006] Fabrice P. Cordelieres (2006). Jacop (just another colocalization plugin).

[Fawcett, 2006] Fawcett, T. (2006). An introduction to roc analysis. pattern recognition letters 27. *861-874*.

[Fay, 1997] Fay, F. (1997). Quantitative digital analysis of diffuse and concentrated nuclear distributions of nascent transcripts, sc35 and poly(a). *Experimental Cell Research*, 231(1):27–37.

[Fletcher et al., 2010] Fletcher, P. A., Scriven, D. R., Schulson, M. N., and Moore, E. D. (2010). Multi-image colocalization and its statistical significance. *Biophysical Journal*, 99(6):1996–2005.

[Frank, 2006] Frank, J. (2006). *Three-dimensional electron microscopy of macromolecular assemblies: Visualization of biological molecules in their native state*. Oxford University Press, Oxford and New York, 2 edition.

[Frank et al., 1996] Frank, J., Radermacher M., Penczek P., Zhu J., Li Y., Ladjadj M., and Leith A. (1996). Spider and web: processing and visualization of images in 3d electron microscopy and related fields. *J Struct Biol 116*, pages 190–199.

[Gelfand et al., 2010] Gelfand, A. E., Fuentes, M., Guttorp, P., Diggle, P., and Fitzmaurice, G. (2010). *Spatial Statistics*. Taylor & Francis Group.

[Georgiev et al., 2011] Georgiev, P., Chlamydas, S., and Akhtar, A. (2011). Drosophila dosage compensation: Males are from mars, females are from venus. *Fly*, 5(2):148–155.

[Gonçalves, 2009] Gonçalves, M. S. T. (2009). Fluorescent labeling of biomolecules with organic probes. *Chemical Reviews*, 109(1):190–212.

[Gonzalez and Woods, 2008] Gonzalez, R. C. and Woods, R. E. (2008). *Digital image processing*. Prentice Hall, Upper Saddle River and N.J, 3 edition.

[Gué et al., 2006] Gué, M., Sun, J.-S., and Boudier, T. (2006). Simultaneous localization of mll, af4 and enl genes in interphase nuclei by 3d-fish: Mll translocation revisited. *BMC Cancer*, 6(1):20.

[Gustafsson, 2005] Gustafsson, M. G. L. (2005). Nonlinear structured-illumination microscopy: Wide-field fluorescence imaging with theoretically unlimited resolution. *Proceedings of the National Academy of Sciences*, 102(37):13081–13086.

[Hall and Patwardhan A., 2004] Hall, R. and Patwardhan A. (2004). A two step approach for semi-automated particle selection from low contrast cryo-electron micrographs. *J Struct Biol 145*, pages 19–28.

[Hansen, 1990] Hansen, L. P. S. (1990). Neural network ensembles. pattern analysis and machine intelligence. *IEEE Transactions on 12*, pages 993–1001.

[Hastie et al., 2009] Hastie, T., Tibshirani, R., and Friedman, J. H. (2009). *The elements of statistical learning: Data mining, inference, and prediction*. Springer, New York, 2 edition.

[Helmuth et al., 2010] Helmuth, J. A., Paul, G., and Sbalzarini, I. F. (2010). Beyond co-localization: inferring spatial interactions between sub-cellular structures from microscopy images. *BMC Bioinformatics*, 11(1):372.

[Huang, 2004] Huang, Z. P. P. (2004). Application of template matching technique to particle detection in electron micrographs. *J Struct Biol 145*, pages 29–40.

[Iannuccelli et al., 2010] Iannuccelli, E., Mompart, F., Gellin, J., Lahbib-Mansais, Y., Yerle, M., and Boudier, T. (2010). Nemo: a tool for analyzing gene and chromosome territory distributions from 3d-fish experiments. *Bioinformatics*, 26(5):696–697.

[Illian et al., 2008] Illian, J., Penttinen, A., Stoyan, H., and Stoyan, D. (2008). *Statistical analysis and modelling of spatial point patterns*. John Wiley, Chichester and England and and Hoboken and NJ.

[Jaskolski et al., 2005] Jaskolski, F., Mulle, C., and Manzoni, O. J. (2005). An automated method to quantify and visualize colocalized fluorescent signals. *Journal of Neuroscience Methods*, 146(1):42–49.

[K. Bache and M. Lichman, 2013] K. Bache and M. Lichman (2013). Uci machine learning repository.

[Kovesi, 1997] Kovesi, P. (1997). Symmetry and asymmetry from local phase. *Tenth Australian Joint Converence on Artificial Intelligence, 2-4*.

[Lachmanovich et al., 2003] Lachmanovich, E., Shvartsman, D. E., Malka, Y., Botvin, C., Henis, Y. I., and Weiss, A. M. (2003). Co-localization analysis of complex formation among membrane proteins by computerized fluorescence microscopy: application to immunofluorescence co-patching studies. *Journal of Microscopy*, 212(2):122–131.

[Langlois, 2011] Langlois, R. J. F. (2011). A clarification of the terms used in comparing semi-automated particle selection algorithms in cryo-em. *J Struct Biol 175*, pages 348–352.

[Leidig et al., 2012] Leidig, C., Bange G., Kopp J., Amlacher S., Aravind A., Wickles S., Witte G., Beckmann R., and Sinning I. (2012). Structural characterization of a eukaryotic chaperoneÂŮthe ribosome-associated complex. *Nat Struct Mol Biol*.

[Lerner B et al., 2007] Lerner B, Koushnir L, and Yeshaya J (2007). Segmentation and classification of dot and non-dot-like fluorescence in situ hybridization signals for automated detection of cytogenetic abnormalities. *IEEE Trans Inf Technol Biomed*, 11(4):443–449.

[Li, 2004] Li, Q. (2004). A syntaxin 1, g o, and n-type calcium channel complex at a presynaptic nerve terminal: Analysis by quantitative immunocolocalization. *Journal of Neuroscience*, 24(16):4070–4081.

[Ljosa et al., 2012a] Ljosa, V., Sokolnicki, K. L., and Carpenter, A. E. (2012a). Annotated high-throughput microscopy image sets for validation. *Nature Methods*, 9(7):637.

[Ljosa et al., 2012b] Ljosa, V., Sokolnicki, K. L., and Carpenter, A. E. (2012b). Annotated high-throughput microscopy image sets for validation. *Nature Methods*, 9(7):637.

[Ludtke et al., 1999] Ludtke, S. J., Baldwin, P. R., and Chiu, W. (1999). Eman: Semiautomated software for high-resolution single-particle reconstructions. *Journal of Structural Biology*, 128(1):82–97.

[Maanders et al., 1993] Maanders, E. M. M., VERBEEK, F. J., and ATEN, J. A. (1993). Measurement of co-localization of objects in dual-colour confocal images. *Journal of Microscopy*, 169(3):375–382.

[Mahy, 2002] Mahy, N. L. (2002). Spatial organization of active and inactive genes and noncoding dna within chromosome territories. *The Journal of Cell Biology*, 157(4):579–589.

[Mallick et al., 2004] Mallick, S., Zhu, Y., and Kriegman, D. (2004). Detecting particles in cryo-em micrographs using learned features. *J Struct Biol 145*, pages 52–62.

[McManus et al., 2006] McManus, K. J., Stephens, D. A., Adams, N. M., Islam, S. A., Freemont, P. S., and Hendzel, M. J. (2006). The transcriptional regulator cbp has defined spatial associations within interphase nuclei. *PLoS Computational Biology*, 2(10):e139.

[Metropolis and Ulam, 1949] Metropolis, N. and Ulam, S. (1949). The monte carlo method. *Journal of the American Statistical Association*, 44(247):335–341.

[Mika et al.,] Mika, S., Ratsch, G., Weston, J., Scholkopf, B., and Mullers, K. Fisher discriminant analysis with kernels. pages 41–48.

[Minsky, 1988] Minsky, M. (1988). Memoir on inventing the confocal scanning microscope. *Scanning*, 10(4):128–138.

[Møller and Waagepetersen, 2004] Møller, J. and Waagepetersen, R. P. (2004). *Statistical inference and simulation for spatial point processes*, volume 100 of *Monographs on statistics and applied probability*. Chapman & Hall/CRC, Boca Raton.

[Morlet, 1982] Morlet, J. G. A. E. F. D. G. (1982). Wave propagation and sampling theory - part ii: Sampling theory and complex waves. *Geophysics 47*, pages 222–236.

[Murphy, 2001] Murphy, D. B. (2001). *Fundamentals of light microscopy and electronic imaging*. Wiley-Liss, New York.

[Netten et al., 1996] Netten, H., van Vliet L. J., Vrolijk H., Sloos W. C. R., Tanke H. J., and Young I. T. (1996). Fluorescent dot counting in interphase cell nuclei. *Bioimaging, 4: 93–106*.

[Nicholson and Glaeser R.M, 2001] Nicholson, W. and Glaeser R.M (2001). Review: automatic particle detection in electron microscopy. *J Struct Biol 133*, pages 90–101.

[Noordmans et al., 1998] Noordmans, H., van der Kraan, K., van Driel, R., and Smeulders, A. (1998). Randomness of spatial distributions of two proteins in the cell nucleus involved in mrna synthesis and their relationship. *Cytometry*, 33(3):297–309.

[Norousi et al., 2013] Norousi, R., Wickles, S., Leidig, C., Becker, T., Schmid, V. J., Beckmann, R., and Tresch, A. (2013). Automatic post-picking using mappos improves particle image detection from cryo-em micrographs. *Journal of Structural Biology*, 182(2):59–66.

[North, 2006] North, A. J. (2006). Seeing is believing? a beginners' guide to practical pitfalls in image acquisition. *The Journal of Cell Biology*, 172(1):9–18.

[Ogura and Sato C., 2004] Ogura, T. and Sato C. (2004). 344-58. *Journal of Structural Biology*, 2004.

[Ollion et al., 2013] Ollion, J., Cochennec, J., Loll, F., Escude, C., and Boudier, T. (2013). Tango: a generic tool for high-throughput 3d image analysis for studying nuclear organization. *Bioinformatics*, 29(14):1840–1841.

[Otsu, 1979] Otsu, N. (1979). A threshold selection method from gray-level histograms. ieee transactions on systems, man, and cybernetics smc-9. *62-66*.

[Pepperkok and Ellenberg, 2006] Pepperkok, R. and Ellenberg, J. (2006). High-throughput fluorescence microscopy for systems biology. *Nature Reviews Molecular Cell Biology*, 7(9):690–696.

[Platt et al., 1988] Platt, W. J., Evans, G. W., and Rathbun, S. L. (1988). The population dynamics of a long-lived conifer (pinus palustris). *The American Naturalist 131*, pages 491–525.

[Quinlan, 1986] Quinlan, J. (1986). Induction of decision trees. *Machine Learning 1*, pages 81–106.

[Raimondo et al., 2005] Raimondo, F., Gavrielides, M., Karayannopoulou, G., Lyroudia, K., Pitas, I., and Kostopoulos, I. (2005). Automated evaluation of her-2/neu status in breast tissue from fluorescent in situ hybridization images. *IEEE Transactions on Image Processing*, 14(9):1288–1299.

[Ram et al., 2010] Ram, S., Rodriguez, J. J., and Bosco, G. (2010). Segmentation and classification of 3-d spots in fish images. In *2010 IEEE Southwest Symposium on Image Analysis & Interpretation (SSIAI)*, pages 101–104. IEEE.

[Rasband, 1997] Rasband, W. S. (1997). Imagej, u. s. national institutes of health, bethesda, maryland, usa. *ImageJ*.

[Ripley, 1976] Ripley, B. D. (1976). The second-order analysis of stationary point processes. *Journal of Applied Probability*, 1976(13):255–266.

[Ronneberger et al., 2008] Ronneberger, O., Baddeley, D., Scheipl, F., Verveer, P. J., Burkhardt, H., Cremer, C., Fahrmeir, L., Cremer, T., and Joffe, B. (2008). Spatial quantitative analysis of fluorescently labeled nuclear structures: Problems, methods, pitfalls. *Chromosome Research*, 16(3):523–562.

[Roseman, 2003] Roseman, A. (2003). Particle finding in electron micrographs using a fast local correlation algorithm. *Ultramicroscopy 94*, pages 225–236.

[Ruusuvuori et al., 2010] Ruusuvuori, P., Aijo, T., Chowdhury, S., Garmendia-Torres, C., Selinummi, J., Birbaumer, M., Dudley, A. M., Pelkmans, L., and Yli-Harja, O. (2010). Evaluation of methods for detection of fluorescence labeled subcellular objects in microscope images. *BMC Bioinformatics*, 11(1):248.

[Schermelleh et al., 2008] Schermelleh, L., Carlton, P. M., Haase, S., Shao, L., Winoto, L., Kner, P., Burke, B., Cardoso, M. C., Agard, D. A., Gustafsson, M. G. L., Leonhardt, H., and Sedat, J. W. (2008). Subdiffraction multicolor imaging of the nuclear periphery with 3d structured illumination microscopy. *Science*, 320(5881):1332–1336.

[Schermelleh et al., 2010] Schermelleh, L., Heintzmann, R., and Leonhardt, H. (2010). A guide to super-resolution fluorescence microscopy. *The Journal of Cell Biology*, 190(2):165–175.

[Schneider et al., 2012] Schneider, C. A., Rasband, W. S., and Eliceiri, K. W. (2012). Nih image to imagej: 25 years of image analysis. *Nat Meth*, 9(7):671–675.

[Shaikh et al., 2008] Shaikh, T., Gao H., Baxter WT., Asturias F.J, Boisset N., Leith A., and Frank J. (2008). Spider image processing for single-particle reconstruction of biological macromolecules from electron micrographs. *Nat Protoc 3*, pages 1941–1974.

[Sheppard and Shotton, 1997] Sheppard, C. and Shotton, D. (1997). *Confocal laser scanning microscopy*, volume 38 of *Microscopy handbooks*. Springer, Oxford and UK and New York and NY and USA.

[Shiels et al., 2007] Shiels, C., Adams, N. M., Islam, S. A., Stephens, D. A., and Freemont, P. S. (2007). Quantitative analysis of cell nucleus organisation. *PLoS Computational Biology*, 3(7):e138.

[Smal et al., 2010] Smal, I., Loog, M., Niessen, W., and Meijering, E. (2010). Quantitative comparison of spot detection methods in fluorescence microscopy. *IEEE Transactions on Medical Imaging*, 29(2):282–301.

[Soper et al., 1917] Soper, H., Young, A., Cave, B., and LEE, A. (1917). On the distribution of the correlation coefficient in small samples. *Biometrika*, 11(4):328–413.

[Sorzano et al., 2009] Sorzano, C., Recarte E, Alcorlo M., Bilbao-Castro JR., San-Martin C., Marabini R., and Carazo JM. (2009). Automatic particle selection from electron micrographs using machine learning techniques. *J Struct Biol 167*, pages 252–260.

[Trabesinger et al., 2001] Trabesinger, W., Hecht, B., Wild, U. P., Schütz, G. J., Schindler, H., and Schmidt, T. (2001). Statistical analysis of single-molecule colocalization assays. *Analytical Chemistry*, 73(6):1100–1105.

[Turin, 1960] Turin, G. (1960). An introduction to matched filters. *IEEE Transactions on Information Theory*, 6(3):311–329.

[Vapnik, 1995] Vapnik, V. (1995). The nature of statistical learning theory. *Springer, New York*.

[Volkmann, 2004] Volkmann, N. (2004). An approach to automated particle picking from electron micrographs based on reduced representation templates. *J Struct Biol 145*, pages 152–156.

[Voss et al., 2009] Voss, N., Yoshioka C.K., Radermacher M., Potter C.S., and Carragher B. (2009). Dog picker and tiltpicker: software tools to facilitate particle selection in single particle electron microscopy. *J Struct Biol 166*, pages 205–213.

[Weinberger et al., 2006] Weinberger, K., Blitzer, J., and Saul, L. (2006). Distance metric learning for large margin nearest neighbor classification. *Advances in Neural Information Processing Systems*, 18:1473–1480.

[Wichard, 2006] Wichard, J. (2006). Model selection in an ensemble framework. *p. 2187-2192, Neural Networks, 2006. IJCNN '06. International Joint Conference on, pp. 2187-2192.*

[Woolford and Hankamer G. Ericksson, 2007] Woolford, D. B. and Hankamer G. Ericksson (2007). The laplacian of gaussian and arbitrary z-crossings approach applied to automated single particle reconstruction. *J Struct Biol 159*, pages 122–134.

[Worz et al., 2010] Worz, S., Sander, P., Pfannmoller, M., Rieker, R. J., Joos, S., Mechtersheimer, G., Boukamp, P., Lichter, P., and Rohr, K. (2010). 3d geometry-based quantification of colocalizations in multichannel 3d microscopy images of human soft tissue tumors. *IEEE Transactions on Medical Imaging*, 29(8):1474–1484.

[Wright and Wright, 2002] Wright, S. J. and Wright, D. J., editors (2002). *Cell Biological Applications of Confocal Microscopy*. Methods in Cell Biology. Elsevier.

[Xiaobo Zhou and Wong, 2006] Xiaobo Zhou and Wong, S. (2006). Informatics challenges of high-throughput microscopy. *IEEE Signal Processing Magazine*, 23(3):63–72.

[Young, 2004] Young, D. W. (2004). Quantitative signature for architectural organization of regulatory factors using intranuclear informatics. *Journal of Cell Science*, 117(21):4889–4896.

[Zhao et al., 2006] Zhao, Y. H., Liao, X. Z., Cheng, S., Ma, E., and Zhu, Y. T. (2006). Simultaneously increasing the ductility and strength of nanostructured alloys. *Advanced Materials*, 18(17):2280–2283.

[Zhu et al., 2004] Zhu, Y., Carragher B., Glaeser R.M, Fellmann D., Bajaj C., Bern M., Mouche F., de Haas F., Hall R.J, Kriegman D.J, Ludtke S.J, and Mallick S.P (2004). Automatic particle selection: results of a comparative study. *Journal of Structural Biology*, 2004:3–14.

Eidesstattliche Versicherung

Hiermit erklare ich an Eides statt, dass die Dissertation von mir selbstständig, ohne unerlaubte Beihilfe angefertigt ist.

Heidelberg, den 06.11.2013 Ramin Norousi